DOCKET NO.: TRIS-1001CON **PATENT**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Meisner et al.**

Serial No.: **10/664,565** Group Art Unit: **2625**

Filed: **September 18, 2003** Examiner: **Azarian, Seyed H.**

For: **AUGMENTED REALITY TECHNOLOGY**

Certificate of Mailing
I, Rebekah L. Mitchell, hereby certify that this correspondence is being deposited with the U.S. Postal Service in an envelope addressed to the Assistant Commissioner for Patents, Washington, DC 20231 on the date shown below.

On: January 17, 2006

By: _____
Rebekah L. Mitchell (Person Mailing Correspondence)

Honorable Commissioner of
Patents and Trademarks
Washington, D.C. 20231

Sir:

RULE 131 DECLARATION

I, Walter Donnelly, hereby declare:

1. I am the Chief Executive Officer of the TriSen Augmented Reality Company, which is the assignee of the above-captioned utility patent application. I am also one of the three inventors named in the application.
2. I am executing this Declaration on behalf of the TriSen Augmented Reality Company as it is not presently possible to locate all three coinventors.
3. The document that is attached hereto as Exhibit A is known to the TriSen Augmented Reality Company and me to have been drafted by my coinventor, Jeffrey Meisner, prior to or on August 6, 1995 and is titled: "Image Transformations in a Videometric Tracking

System." This was a confidential internal document belonging to TriSen Systems and was not disseminated to the public. It describes the original design specification and architecture for the TriSen augmented reality system.

4. The document that is attached hereto as Exhibit B is known to the TriSen Augmented Reality Company and me to have been drafted by Mr. Meisner prior to or on August 1, 1995 and is titled: "Analysis of Error in the Videometric Tracking System" This also was a confidential internal document of TriSen Systems and was not disseminated to the public. This is the original error analysis document for the TriSen Videometric tracker.
5. The document that is attached hereto as Exhibit C is a progress report that is known to the TriSen Augmented Reality Company and me to have been prepared by TriSen and used in a conference that was held on November 2, 1995 to report progress, in confidence, to the current prime contractor, Honeywell.
6. The document that is attached hereto as Exhibit D is a paper copy of an e-mail message dated March 26, 1996 that is known to the TriSen Augmented Reality Company and me to have been sent from TriSen to Honeywell reporting interim status in converting the code for the working TriSen tracker from the Forth programming language to C.
7. The document that is attached hereto as Exhibit E is an interface specification that is known to the TriSen Augmented Reality Company and me to have been prepared by the inventors prior to or on July 9, 1996 for the second VFE design, which used a parallel port to communicate with the wearable computer.
8. The document that is attached hereto as Exhibit F is known to the TriSen Augmented Reality Company to have been drafted on or before October 30, 1995 and is the inventors' original hand drawn design for the original VFE, including the equations used in the programmable gate array logic device, showing some development already completed.
9. All of the attached Exhibits were drafted within, and the events referred to occurred within, the United States of America.

10. The attached documentation, and Exhibits A, B, C and D in particular, demonstrates that the inventors prior to August 2, 1996 conceived a method for tracking the position and orientation of an object as described in independent claim 1, specifically that included steps of (a) scanning across an object to detect fiducials, wherein a video run is formed by a scan; (b) clumping video runs to detect a pattern of fiducials; (c) acquiring estimated values for a set of tracking parameters by comparing a detected pattern of fiducials to a reference pattern of fiducials; and (d) iterating the estimated values for the set of tracking parameters until the detected pattern of fiducials match the reference pattern of fiducials to within a desired convergence.
11. Exhibit C, Page 8 describes a process where a video run is formed by a scan, and wherein video runs are clumped to detect a pattern of fiducials. The "tracking correspondence and estimation" on page 8 of this document describes a process of acquiring estimated values for a set of tracking parameters by comparing a detected pattern of fiducials to a reference pattern of fiducials.
12. Exhibit F describes in detail a process of acquiring estimated values and iterating the estimated values for the set of tracking parameters until the detected pattern of fiducials match the reference pattern of fiducials to within a desired convergence.
13. The attached documentation, and Exhibits A, B, C, D and F in particular, demonstrates that the inventors prior to August 2, 1996 conceived a method for augmenting reality as described in independent claim 11, including steps of (a) tracking the position and orientation of a pattern of fiducials on an object with a self-contained, mobile system; (b) processing virtual information stored in a computer memory of said system according to the position and orientation of the object; and (c) presenting the virtual information with real information to a user in near real time with the system.

14. The user interface for receiving input and presenting augmented information to a user, and which includes a head mounted (mobile) display on which the computer presents the augmented information is described on page 3 of Exhibit C.
15. Exhibit C, Page 8 describes a process where a video run is formed by a scan, and wherein video runs are clumped to detect a pattern of fiducials. The "tracking correspondence and estimation" on page 8 of this document describes a process of acquiring estimated values for a set of tracking parameters by comparing a detected pattern of fiducials to a reference pattern of fiducials.
16. Exhibit F describes in detail a process of acquiring estimated values and iterating the estimated values for the set of tracking parameters until the detected pattern of fiducials match the reference pattern of fiducials to within a desired convergence.
17. The attached documentation, and Exhibits A, B and C in particular, demonstrates that the inventors prior to August 2, 1996 conceived a method for augmenting reality as described in independent claim 26, comprising steps of (a) using a sensor to provide at least one signal that is indicative of a pattern of fiducials on an object; (b) processing the signal to locate the fiducials; (c) determining a relative position and orientation of the sensor with respect to the object by comparing the locations of said fiducials to a known reference pattern; (d) providing virtual information to a user in substantial registration with real information based on the relative position and orientation determined in step (c), and wherein the method is performed so as to provide the virtual information to said user in near real-time.
18. More specifically, the optical sensor that is referenced in several of the claims is described in several of the Exhibits as a video camera. For example, Exhibit A, p. 1 mentions a video camera in several places. Exhibit C lists the specific hardware that was used in the invention as it was contemplated on November 2, 1995.

19. Exhibit C, Page 8 describes a process where a video run is formed by a scan, and wherein video runs are clumped to detect a pattern of fiducials. The "tracking correspondence and estimation" on page 8 of this document describes a process of acquiring estimated values for a set of tracking parameters by comparing a detected pattern of fiducials to a reference pattern of fiducials.
20. Exhibit A, p. 1 specifically describes an image containing a number of fiducial markings, and states that the fiducial markings are on an object plane and that the positions of the fiducial markings are known with certainty.
21. The calculation of angular parameters including phi, theta and psi, wherein said position of said at least one sensor with respect to said pattern of fiducials is defined by linear parameters including X-bar, Y-bar, and distance, said hard fiducial pattern being used to identify said phi, theta, psi and distance parameters and said soft fiducial pattern being used to identify said x-bar and y-bar parameters is described in Exhibits A and B.
22. The step of processing virtual information stored in a computer memory according to the position and orientation of the object is described in Exhibit C, p. 9. P. 4 of this document describes the presentation of virtual information with real information to a user in near real-time.
23. A prototype that would be read upon by the independent claims presently pending in the above captioned application was manufactured, reduced to practice and demonstrated to Honeywell in July 1996. It was later delivered by TriSen to the Boeing Corp. in mid-1996. The undersigned is still attempting to recover documentation relating to this prototype.
24. TriSen continued to develop and refine the technology in the period of time between the conception described above and the filing of the provisional patent application upon which the above captioned patent application claims priority. The pace of such development was complicated somewhat by the death of the president of TriSen and the departure of Mr.

Meisner from the company. However, the company remained diligent in reducing the invention to practice throughout this period.

25. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that the statements were made with the knowledge that willful false statements and the like made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Respectfully submitted,

TRISEN AUGMENTED REALITY COMPANY

By Walter Donnelly
Chief Executive Officer

Date: 1/15/06

EXHIBIT A

(A)

Image Transformations in a Videometric Tracking System

J. Meisner, TriSen Systems

August 6, 1995

In a *videometric* tracking system, an object plane defined in 3-space as z=0, is imaged onto the image plane of a video camera. The image plane coordinates are denoted (u,v). The camera is positioned at x_o, y_o, z_o, and is first rotated horizontally in the x-z plane (yaw) by an angle θ, then rotated downwards (in the -y direction) by an angle ϕ (pitch), and then rotated about its own axis by an angle ψ. The resultant image is therefore a function of the markings on the object plane and the six tracking parameters described above. Additionally, the image transformation is dependent on the scale (or equivalently the camera's field of view, or lens focal length), which does not change.

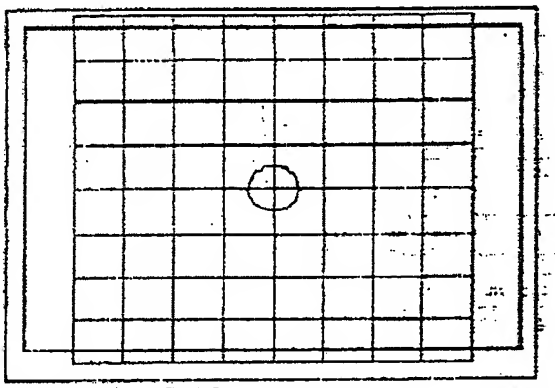


Figure 1. Grid pattern as seen straight on (all rotation angles are zero).

We would like to explore the general character of mappings from one two-dimensional surface to another, and consider solutions to the tracking problem given an image so obtained, containing a number of fiducial markings on the object plane whose positions on the object plane are known with certainty.

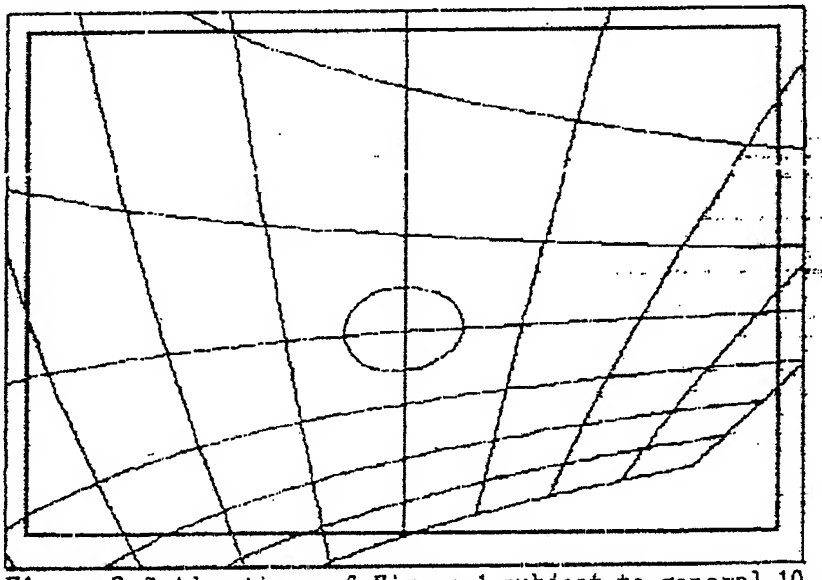


Figure 2. Grid pattern of Figure 1 subject to general 10 D.F. transform. Note that lines in object plane are curved in image.

We will only

consider bi-linear transforms from the (x,y) object plane to the (u,v) image plane. The most general bi-linear transform is described as follows.

$$u = \frac{a'x + b'y + c'}{d'x + e'y + 1} \tag{1}$$

$$v = \frac{a''x + b''y + c''}{d''x + e''y + 1} \tag{2}$$

This very general transform is seen to have 10 degrees of freedom. In Figure 1 a grid pattern is shown as it has been simulated on an object plane (seen straight on, thus without distortion). An arbitrary transform of the form of (1), (2) has been performed. The result is shown in Figure 2. Note that straight lines on the object plane have become curved, a phenomenon that will not occur using distortion-free optics.

As opposed to the 10 D.F. transform considered in (1), (2), a plane projection transform, which describes the image formed by a telescopic lens at a large distance, is described by only six parameters. Setting d'=e'=d"=e"=0, we obtain simply:

$$u = a'x + b'y + c' \tag{3}$$

$$v = a''x + b''y + c'' \tag{4}$$

This therefore approximates a narrow field-of-view lens, in which the ratio of areas between the object plane and the image plane is constant

throughout the image (as can be easily verified by computing the Jacobian of the transform). The grid pattern in Figure 1 has been transformed with a 45° downward view in Figure 3 and with a 45° horizontal rotation followed by a 45° downward rotation in Figure 4. Because the ratio of areas between the object plane and the image plane is constant, there is a two-way ambiguity in the interpretation of the transformed image. In Figure 3, for example, it is impossible to tell if ϕ=+45° or ϕ=-45°, we are only left to observe the compression of the vertical axis according to the cosine of ϕ!

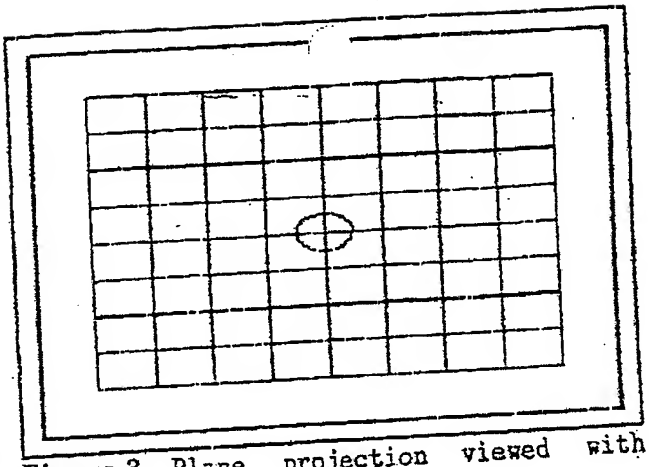


Figure 3. Plane projection viewed with ϕ=45°.

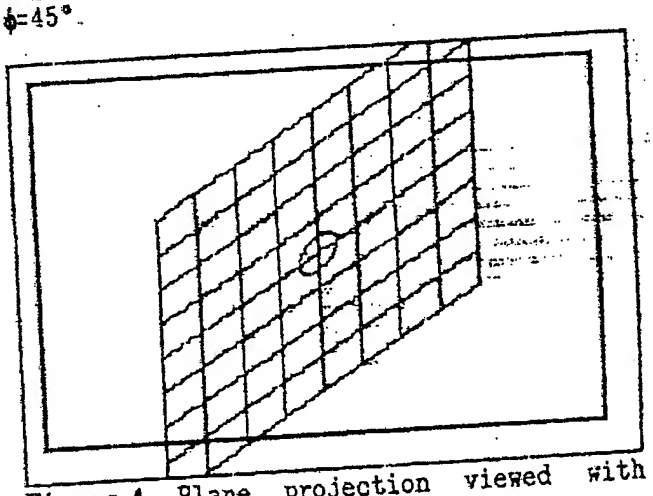


Figure 4. Plane projection viewed with θ=45°, ϕ=45°.

This flaw is not present in an image obtained using a finite projection, obtained with a camera having a substantial field of view. For instance in Figure 5, the object of Figure 1 is observed, again with a downward view of ϕ=45°, using a video camera whose horizontal field-of-view is 90°. The change in the projection of areas, from top to bottom, is quite apparent, thus resolving the ambiguity in the sign of ϕ obtained by observing the image of the known object.

The finite projection transformation is indistinguishable from the arbitrary bilinear transformation of (1), (2), except for constraints on the values of the ten parameters. In particular, the denominators of the expressions for u and v are required to be identical, a condition which insures that lines in the object plane will be transformed to lines in the image plane.

$$u = \frac{a'x + b'y + c'}{dx + ey + 1} \tag{5}$$

$$v = \frac{a''x + b''y + c''}{dx + ey + 1} \tag{6}$$

Although (5), (6) contain 8 parameters, there are only six actual degrees of freedom in the transformation, corresponding to the three positional parameters and the three rotational parameters we are tracking. In terms of the tracking parameters, the 8 coefficients in (5), (6) are given by:

$$\begin{aligned}
a' &= g(\cos\psi\cos\theta + \sin\psi\sin\phi\sin\theta) \\
b' &= g(\sin\psi\cos\phi) \\
c' &= g(x_0(\cos\psi\cos\theta - \sin\psi\sin\phi\sin\theta) \\
&\quad - y_0\sin\psi\cos\phi \\
&\quad + z_0(\sin\psi\sin\phi\cos\theta - \cos\psi\sin\theta)) \\
a'' &= g(-\sin\psi\cos\theta + \cos\psi\sin\phi\sin\theta) \\
b'' &= g\cos\psi\cos\phi \\
c'' &= g(x_0(\sin\psi\cos\theta - \cos\psi\sin\phi\sin\theta) \\
&\quad - y_0\cos\psi\cos\phi \\
&\quad + z_0(\sin\psi\sin\theta + \cos\psi\sin\phi\cos\theta)) \\
d &= g\cos\phi\sin\theta \\
e &= -g\sin\theta
\end{aligned} \tag{7}$$

where:

$$g \triangleq \frac{1}{-x_0\cos\phi\sin\theta + y_0\sin\phi + z_0\cos\phi\cos\theta}$$

$X_0 = X_1 - Z_0 \tan\theta$

$Y_0 = Y_1 + Z_0 \tan\phi \sec\theta$

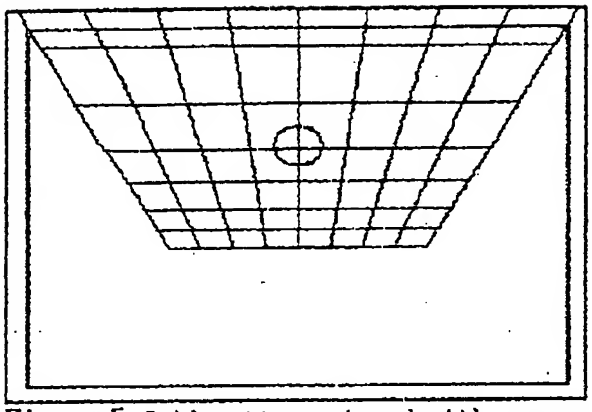


Figure 5. Grid pattern viewed with camera having 90° horizontal FOV, with ϕ=45°.

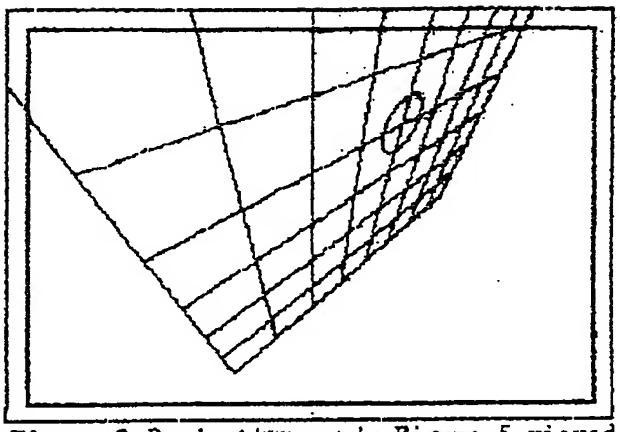


Figure 6. Projection as in Figure 5 viewed from θ=45°, ϕ=45°.

Although observation of the transformation (as viewed, for instance, in Figure 6) unambiguously determines the six tracking parameters, there is still an ambiguity when only observing the position of only three fiducial points. Even though three points on the image plane, having six degrees of freedom, are sufficient to solve for the six tracking parameters, there are still two solutions that satisfy the transformed positions of only three points. Thus a tracker observing only three fiducial points is capable of "latching on" to the wrong solution and diverging from the correct estimation. This will not occur when four or more fiducial are within the camera's field-of-view.

Since four fiducials are necessary to unambiguously determine the tracking parameters, it may be possible to perform an estimation algorithm not on the above model having 6 degrees of freedom, but rather on the 8 degree of freedom aphysical model. The 8 D.F. model is also described by (5), (6), however the constraints of (7) are no longer imposed. While this model does not correspond to a physical paradigm, it is simpler than the 10 D.F. model of (1), (2), and guarantees that lines in the object plane are transformed to lines in the image plane. An estimation of the 8 parameters of (5), (6), can be followed by collapsing the estimation into a 6 D.F. model consistent with (7), thus retrieving the tracking parameters. If four fiducial points are being observed and the eight parameters of (5), (6) are then solved, it is expected that the derivative solution of the six tracking parameters

would match those obtained by an estimation algorithm based on only the six tracking parameters. However the initial estimation procedure may be simpler since trigonometric functions are no longer involved in the 8 D.F. model.

Parameter Estimation using the Gradient Descent Method

Let us suppose that we have observed at least four fiducial points in the image plane at coordinates u_1, v_1; u_2, v_2; etc. Suppose that the positions of the fiducial marks on the object plane are known to be x_1, y_1; x_2, y_2; etc. Then, for a given 8 D.F. transformation, we would have expected to observed these fiducials at positions:

$$\hat{u}_i = \frac{a'x_i + b'y_i + c'}{dx_i + ey_i + 1} \tag{8}$$

$$\hat{v}_i = \frac{a''x_i + b''y_i + c''}{dx_i + ey_i + 1} \tag{9}$$

Then we would find an estimate of the eight parameters which minimized the net cost function Q, defined as the sum of the squares of the residuals.

$$Q = \sum_i (\hat{u}_i - u_i)^2 + (\hat{v}_i - v_i)^2 \tag{10}$$

Then the gradient of Q is found as the 8 element vector formed of the partial derivatives of Q with respect to each parameter.

$$\frac{\partial Q}{\partial a'} = \sum_i 2 \frac{x_i}{dx_i + ey_i + 1} (\hat{u}_i - u_i) \qquad (11)$$

$$\frac{\partial Q}{\partial b'} = \sum_i 2 \frac{y_i}{dx_i + ey_i + 1} (\hat{u}_i - u_i) \qquad (12)$$

$$\frac{\partial Q}{\partial c'} = \sum_i \frac{2}{dx_i + ey_i + 1} (\hat{u}_i - u_i) \qquad (13)$$

$$\frac{\partial Q}{\partial a''} = \sum_i 2 \frac{x_i}{dx_i + ey_i + 1} (\hat{v}_i - v_i) \qquad (14)$$

$$\frac{\partial Q}{\partial b''} = \sum_i 2 \frac{y_i}{dx_i + ey_i + 1} (\hat{v}_i - v_i) \qquad (15)$$

$$\frac{\partial Q}{\partial c''} = \sum_i \frac{2}{dx_i + ey_i + 1} (\hat{v}_i - v_i) \qquad (16)$$

$$\frac{\partial Q}{\partial d} = \sum_i \frac{2x_i}{dx_i + ey_i + 1} [(u_i - \hat{u}_i)\hat{u}_i + (v_i - \hat{v}_i)\hat{v}_i] \qquad (17)$$

$$\frac{\partial Q}{\partial e} = \sum_i \frac{2y_i}{dx_i + ey_i + 1} [(u_i - \hat{u}_i)\hat{u}_i + (v_i - \hat{v}_i)\hat{v}_i] \qquad (18)$$

Multiplying the negative of the gradient of the cost function by a gain parameter, and adding that quantity to the parameter vector, will result in a iterative procedure which will solve the 8 D.F. estimation problem in a least-squares sense.

EXHIBIT B

(B)

Analysis of Error in the *Videometric* Tracking System

J. Meisner, TriSen Systems

August 1, 1995

A *videometric* head tracking system has been proposed in which a video camera mounted on a helmet will observe fiducial marks on a planar board in order to estimate head position for the purpose of driving a head mounted display (HMD) in order to superimpose computer-generated symbols to appear on the same plane as the fiducial marks The pertinent error measure is **not** the error in the measurement of head position or orientation, but rather the **net** error in the position of the computer-generated symbols as they are superimposed on the plane at which the fiducials reside (hereafter: **object plane**).

The ensuing analysis will consider four different coordinate systems which will now be defined. There is a **fixed** three-dimensional coordinate system denoted x, y, z, relative to which fiducials (and intended symbol placement) exist in an absolute sense. The position of the (nominally) planar surface is defined by $z=0$. The y coordinate may be thought of as vertical, and the x coordinate as horizontal, along the surface

A **translated and rotated** three dimensional coordinate system, **denoted** x', y', z', is defined by the position of the pupil of **the camera** mounted on the helmet The origin of the primed **coordinate** system in the fixed coordinate system is located at x_o, y_o, z_o. **Following** translation to that point, a rotation given by the orthogonal matrix R yields the primed coordinates. The two three-dimensional coordinate systems are thus related by:

$$\vec{r}' = \mathbf{R}(\vec{r} - \vec{r}_0)$$
$$\vec{r} = \vec{r}_0 + \mathbf{R}^{-1}\vec{r}'$$

1

Then there are two two-dimensional coordinate systems to consider. One is simply the coordinate system of the object plane (on which the fiducials reside) which uses the coordinates x (horizontal) and y (vertical). These are simply the same x and y used in the fixed 3-D coordinate system, at which the object plane was stipulated to exist at z=0.

Finally there is a two-dimensional coordinate system corresponding to the image received by the camera. For the purposes of the following analysis, we will consider the pupil of the eye to be located at the position of the pupil of the video camera, thus this coordinate system is also directly mapped to the visual field of the human. The coordinates of this system are labeled u, v, where u is (for upright head positions) approximately horizonal (parallel to x´), and v is approximately vertical (parallel to y´).

We define the scale of this coordinate system such that the tangent (not sine) of a deflection angle is given by $\sqrt{(u^2+v^2)}$. In particular, using a lens of focal length f, on the physical image plane x",y", where x" and y" are anti-parallel to x´,y´, (due to image inversion), u and v are defined by u=x"/f and v=y"/f. Then, if an object at a point in space (in the rotated 3-D coordinate system) exists at a position x´,y´,z´, it will be imaged at u,v, such that:

$$x' = Lu$$
$$y' = Lv$$
$$z' = -L$$

2

L, above, is a parameter expressing the distance between the camera pupil and a plane normal to the axis of the camera which contains the point x',y',z'. Simply given an image in the u,v coordinate system, L is *a priori* unknown. However given a transformation between the fixed and the primed coordinate systems, and knowing that an imaged point is located on the planar board at $z=0$, L can be solved for, thus obtaining the absolute location of that imaged point using 2 and 1.

A few physical details related to the image plane will be introduced at this point The coordinates u and v will be considered to run from $-\infty$ to ∞, thus covering a 180° field of view. Within this space, a physical video camera will have a finite field of view in which u and v are bounded by $-l_{FOV}$ and $+l_{FOV}$. Of course most TV cameras have a 4:3 aspect ratio, however that detail will be ignored (or perhaps averaged) in the specification of l_{FOV}. Even though the field of view of the camera itself is limited, we will be considering the position of a point of interest at further locations on the u,v plane.

Now, a video camera with a resolution of R pixels across the above specified field of view, will have a pixel size in the image plane of $\Delta u=\Delta v=2l_{FOV}/R$. If we consider the measurement error e, in the u,v coordinate system, to be 1 pixel r.m.s. in each u and v, then $e=2\sqrt{2}l_{FOV}/R$ On the basis of this relationship, we can define an effective resolution R_{eff} for a camera which has a net r m s. error e in the determination of the position of an imaged fiducial point. For a camera with no appreciable lens field distortion, fiducial image sizes of a few pixels, and effective centroiding, we would expect R_{eff} to be close to (or somewhat less than) the actual hardware pixel resolution of the camera.

Analysis of Positioning Error in the Image Plane

Given a field containing several fiducials in which we assume a nominal measurement error, and a point of interest P in that field whose location is specified with respect to those fiducials, let us determine the resultant error in the determination (or placement) of the point of interest P. Note that this discussion only pertains to the positional errors as seen in the image plane. In other words, we are only considering the positional errors as they would appear with respect to the television image returned by the video camera, with no regards to the consequent positional error in the object plane (or in 3-space). The propagation of these errors measured in the image plane to positioning errors in the object plane will be considered at a later time.

In order to determine the magnitude of positioning errors due to imprecise measurement of the fiducial positions, we will assume that there are N fiducials (where $N \geq 3$) in the field of view of the camera, and that the area defined by this array of fiducials is centered about $u=0$, $v=0$, and of approximate extent l in both u and v. Since the magnitude of errors is not sensitive to the specific positioning of fiducials relative to P, we can just as well consider the special case in which there are 4 fiducials

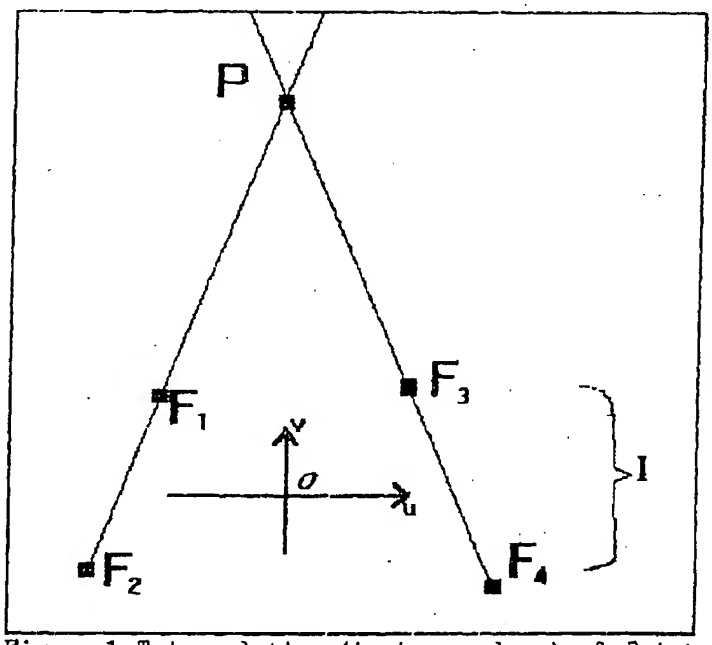


Figure 1. Triangulation (in image plane) of *Point of Interest* P, using four fiducials, possible due to fact that lines in object plane transform to lines in image plane.

in a pattern in which the line defined by one pair of fiducials, F_1 and F_2, and the line defined by the other two fiducials, F_3 and F_4, conveniently intersect at the point of interest P. In other words, P is *triangulated* by the lines formed by each pair of fiducials as shown in Figure 1. This is an appropriate model to apply to the eventual errors that would be obtained in the object plane, since a point which is triangulated by two such lines in the image plane, would also be triangulated in the object plane by two such lines also defined by the two pairs of fiducials as they exist in the object plane (since lines in the object plane are imaged as lines)

As can be seen, the geometry specified in Figure 1 corresponds to the point of interest lying well outside the camera field-of-view: the case of positional *extrapolation* We will limit our error analysis to such a geometry since extrapolation corresponds to the 'worst case" error result.

So let us assume that each of these four fiducial positions are determined with a r.m.s. error in the image plane of e Assuming isotropic errors, that would imply errors in u and v, each of magnitude $e\sqrt{2}/2$ Then it can be seen that the error $\Delta\phi$ in the orientation of either line triangulating P in Figure 1 would be given by:

$$\Delta\phi = \frac{e}{l} \qquad 3$$

Additionally, the error Δp in the positioning of that line, in the direction normal to the line itself, would be:

$$\Delta p = \frac{e}{2} \qquad 4$$

Then it can be seen that the net error in the triangulation of P using lines with errors given by 3 and 4 would be, in the direction normal to the line connecting P to the origin:

$$\Delta p_1 = e\frac{\sqrt{2}}{2}\left(\frac{1}{2} + \frac{|P|}{1}\right) \qquad 5$$

where |P| denotes the distance of P from the origin in the image plane. The error in the triangulation of P parallel to the direction of the line connecting P to the origin, would be:

$$\Delta p_2 = e\frac{\sqrt{2}}{2}\frac{|P|}{1}\left(1 + \frac{|P|}{1}\right) \qquad 6$$

In the extrapolation case where |P|>1, the second error 6 is clearly of greater significance. However the specific contribution of either error towards the resultant positional error in the object plane, will depend on the coordinate transformation, which is dependent on the actual 3-dimensional geometry, not considered in the above examination.

Now, 5 and 6 are based on the triangulation of P using four fiducial positions. For an increased number of fiducials, a positional determination in which the independent errors due to each were averaged would decrease the net error by the square root of N, the number of fiducials employed. Thus, for N fiducials, we would rewrite 5 and 6 as:

$$\Delta p_1 = e\sqrt{\frac{1}{2N}}\left(1 + \frac{|P|}{l}\right) \qquad 7$$

$$\Delta p_2 = e\sqrt{\frac{2}{N}}\frac{|P|}{l}\left(1 + \frac{|P|}{l}\right) \qquad 8$$

The Coordinate Transformation

Let us now look at the coordinate transformation between the two-dimensional image plane coordinate system, and the two-dimensional object plane. This transformation is a result of, and unique to, the specific position and orientation of the video camera in space. That includes three degrees of freedom in position and three degrees of freedom in describing the camera's orientation. However we would like to identify the significant degrees of freedom which generate a change in the object plane error due to errors in the image plane. In that vein, we will find only two significant degrees of freedom.

Consider first the position of the camera's pupil, given by a position x_o, y_o, z_o in space. Of these 3-space coordinates, x and y are parallel to the object plane, and a simultaneous shift in the camera position and the fiducial positions (or the presence of similar fiducials in the new position) will leave the problem of error propagation unchanged. However the coordinate z_o describing the distance between the camera and the physical position of the object plane, clearly is of consequence.

The orientation of the camera in space is given by three angles. Two of these angles describe the direction of the axis of the camera. The third angle describes a rotation of the

camera about that axis. That rotation is equivalent to a rotation of all points in the image plane (as described in the previous section) and has no further significance. The other two angles do alter the coordinate transformation in a significant way. However the qualitative changes in that transformation are clearly dependent on the angle made between the camera's axis and a normal to the object plane, but not on the orientation of the plane which includes the camera's axis and the normal to the object plane So there is only one significant degree of freedom in camera orientation, relative to analysis of the propagation of positional error.

Therefore, we shall, without loss of generality, model the coordinate transformation as being a result of a camera whose pupil is located at x=0, y=0, and $z=z_0$, and whose orientation is altered only by a pointing of the axis of the camera downward (in the -y direction) by the angle ϕ. For this special case, the reverse transformation, from the image plane to the object plane, can be simply written as:

$$x = \frac{u z_0}{v \sin\phi + \cos\phi}$$

$$y = \frac{v z_0 \cos\phi - z_0 \sin\phi}{v \sin\phi + \cos\phi} \qquad 9$$

Then, the sensitivity of errors in the object plane to errors in the image plane, is given by the (partial) derivatives of the coordinate transformation 9 with respect to the image plane coordinates.

Let us consider the point of interest P, to be found on the image plane at a point u,v. Let us call D the distance in

3-space between the camera pupil and the actual point of interest P on the z=0 plane. In terms of u,v and the transformation parameters, D can be expressed as:

$$D = \frac{z_0\sqrt{1+u^2+v^2}}{v\sin\phi + \cos\phi} \qquad 10$$

However we can just view D as an essential parameter expressing the distance between the tracking hardware (camera) and the work.

Then the derivatives of the reverse coordinate transformation, evaluated around u,v the point of interest, can be shown to be given by:

$$\frac{\delta x}{\delta u} = \frac{D}{\sqrt{1+u^2+v^2}} \qquad 11$$

$$\frac{\delta x}{\delta v} = -u\sin\phi\,\frac{D^2}{z_0\,(1+u^2+v^2)} \qquad 12$$

$$\frac{\delta y}{\delta u} = 0 \qquad 13$$

$$\frac{\delta y}{\delta v} = \frac{D^2}{z_0 (1 + u^2 + v^2)} \qquad 14$$

The mixed derivatives (considering that x is nominally parallel to u, and y is nominally parallel to v) given by 12 and 13, are dominated by the other two derivatives, and will henceforth be ignored.

Calculation of Net Error in the Object Plane

Using the error in the image plane calculated earlier in 7 and 8, and using the sensitivity of errors in the object plane to errors in the image plane given by 11 and 14, we can find the net errors as measured in the object plane. Thus we will find the positional error in the placement of computer-generated symbols as they appear on the plane at which their registration is defined.

The worst case net error occurs when the point of interest P is offset from the u,v origin in the same direction as the tilt defined by ϕ. We then obtain the net error in the object plane by multiplying the error in the image plane given by 8, by the derivative of object plane position with respect to image plane position, given by 14 We thus obtain for the error in the object plane ΔL, the following:

$$\Delta L = \Delta p_2 \frac{\delta y}{\delta v} = e\sqrt{\frac{2}{N}}\frac{|P|}{1}\left(1 + \frac{|P|}{1}\right)\frac{D^2}{z_0 (1 + u^2 + v^2)} \qquad 15$$

For simplicity, let us examine the special (but not atypical) case when axis of the tracking camera is normal to the physical object plane: $\phi=0$. Let us call L the physical distance on the object plane between the point that the axis of the camera intersects the object plane and the physical point of interest P. Then 15 can be simplified by applying 10 and the definition of L.

$$\Delta L = \frac{e}{l}\sqrt{\frac{2}{N}}\left(1+\frac{L}{l z_0}\right)L \qquad (\phi=0) \qquad 16$$

So for the case of $\phi=0$, 16, 20 says that the error on the object plane in locating a point of interest P, located a distance L from the point at which the axis of the camera intersects the object plane, is given as the product of three factors times L. The first factor, e/l, in 16, 20, can be viewed as the reciprocal of the number of resolvable points in between fiducial markings. The second factor is on the order of unity, decreasing slightly for a larger number of fiducials in the camera's field of view. The third factor is one plus the *extrapolation coefficient*, defined as the ratio of L to the distance between physical fiducials on the object plane.

As a hypothetical example, consider a video camera with a wide angle lens having a 90° field of view Suppose that the errors in the image plane are about 1 pixel, for a resolution of 500 lines. If the camera is located at 20 cm from the object plane, its field of view, at $\phi=0$, will subtend a square on the object plane of 40 cm size (with each pixel subtending a square of .8 mm) Suppose that a field of 4 fiducial marks covers 3/4 of this size: $lz_0=30$ cm. Then if the point of interest is located at a distance of 50 cm below the camera, we would calculate the net error to be:

$$\Delta L = \frac{.004}{1.5}\sqrt{\frac{2}{4}}\left(1 + \frac{50\,cm}{30\,cm}\right)\cdot 50\,cm = .25\,cm \qquad 17$$

Thus we find a net positioning error of only one quarter of a centimeter. Now imagine that the fiducials are only spaced by 10 cm (instead of 30 cm) thus increasing the extrapolation coefficient L/lz, from 1.67 to 5, and decreasing the number of accurately resolvable points between fiducials 1/e, from 375 to 125. Under this more pessimistic scenario, we would find the net object plane positional error increased to:

$$\Delta L = \frac{.004}{.5}\sqrt{\frac{2}{4}}\left(1 + \frac{50\,cm}{10\,cm}\right)\cdot 50\,cm = 1.7\,cm \qquad 18$$

Qualitatively, we can attribute this unacceptable error result to a combination of the increased extrapolation coefficient due to closely spaced fiducials far from the point of interest, and a camera field of view poorly matched to the narrow field of fiducials. The lesson of this example, is to provide a rich field of fiducials that would largely fill any likely camera field of view, and not be widely separated (relative to their spacing) from the points of interest.

For a field with a high density of fiducials, there can be assumed to be an extent 1, approaching the total field of view of the video camera, given by $2l_{FOV}$. Let us employ the *effective resolution* R_{eff}, defined earlier, for an image plane error e, as:

$$R_{eff} \Delta 2\sqrt{2}\,\frac{l_{FOV}}{e} \qquad 19$$

Then we can write, assuming a high density of fiducials, the estimation error 16, 20 for the case of $\phi=0$, as:

$$\Delta L = \frac{2}{R_{eff}\sqrt{N}}\left(1 + \frac{L}{2 l_{FOV} z_0}\right) L \qquad (\phi=0) \qquad 20$$

Everything else being equal, it can be seen that the error will be inversely proportional to the camera field of view l_{FOV}. What's more, the required number of physical fiducials to achieve a given N within the camera's field of view, will decrease according to l_{FOV}^{-2}.

Regardless of the placement of fiducials, however, a situation in which the distance between the camera and the object plane is allowed to dwindle, will necessarily decrease the physical spacing between observed fiducials. If this is allowed to occur without a corresponding decrease in L, then an increased error due to a large extrapolation coefficient will occur. A reasonable strategy for avoiding this circumstance, is to place the tracking camera as close as possible to, and parallel to the direction of, the eye viewing the computer generated display, so that the human will never be looking in a location too far from the field observed by the video camera.

Expected Performance of Proposed Configurations

In Table I, three systems are summarized. The system designated "A" corresponds to a camera based on the imputer® (VLSI Vision Ltd.) which includes a low-resolution video camera. The lens provides a (horizontal) field-of-view of 35°, requiring a density of fiducials of 200 per square meter in order to be reliable at distances from the camera to the board as small as z_o = 30cm. Case "B" corresponds to a reasonably high resolution camera with a 60° field-of-view, still using a board containing 200 fiducials per square meter. Case "C" represents the implementation of the same camera with a wide angle lens (90°), but in which the density of fiducials is decreased from 200 to 50, possible with the wider lens field-of-view.

Table I

Specifications for Three Proposed Systems

	A	B	C
Pixel Resolution (H x V)	256 x 256	512 x 480	512 x 480
Density of Fiducials ρ (# per square meter)	200	200	50
Field of View (horizontal), θ_{FOV}	35°	60°	90°
l_{FOV}	.315	.577	1.
1 pixel error in image plane, e	.00154	.00144	.0025

In Table II, the performance of each of these systems is evaluated for six different cases. We look at three different camera-board distances z_o, of 60 cm, 30 cm, and 20 cm. In each case, we have explored two different values of L, the distance between the point at which the camera's axis intersects the board, and the point of interest at which the error is evaluated. In each case one value of L is reasonably small, so that the point of interest is near or inside the field-of-view of the camera itself. The other corresponds to a substantial extrapolation, thus revealing the worst case performance. In each example, we have assumed that the axis of the camera is approximately normal to the board.

Table II

Table of Error Performance for Cases A, B, and C

(All lengths in centimeters)

	z_o	L	Case	N	P_0	P_1	P_2	P_3	l	Extrap	$\Delta L/2$
1	60	20	A	21.4					.37	(.90)	(.05)
2			B	72.					70	(.47)	(.01)
3			C	54.					1.22	(.27)	(.02)
4	60	100	A	21.4					.37	4.5	.7
5			B	72.					.70	2.36	.11
6			C	54.					1.22	1.37	.09
7	30	20	A	5.4	.21	.09	.03	.005	.317	1.49	.15
8			B	18.					.675	(.99)	(.03)
9			C	13 5					1.15	(.58)	(.03)
10	30	75	A	5.4	.21	.094	.03	.005	.317	7 9	2.0
11			B	18.					.675	3.7	25
12			C	13 5					1.15	2.2	20
13	20	15	A	2.4	78	57	.31	.09	.23	3.3	.40
14			B	8.	.04	.015	.003		.625	1.20	.04
15			C	6.	.15	.06	.017	.003	1.03	(.72)	(.04)
16	20	50	A	2.4	.78	.57	.31	.09	.23	11.0	3.4
17			B	8.	.04	.015	.003		.625	4.0	.26
18			C	6.	.15	.06	.017	.003	1.03	2.4	.22

Table II shows that system "A' obtains unacceptable errors in the case of substantial extrapolation (lines 10 and 16), especially at smaller values of z_o.

Abundance of Fiducials

In Table II, the column labelled "N" shows the expected number of fiducials that would be seen by the camera, dependent on its field of view θ_{Fov}, the density of fiducials on the board ρ, and the specific distance between the camera and the board z_o. The value of N enters into the determination of the expected object plane positioning error ΔL, due to the increased accuracy in parameter estimation using multiple independent measurements. N, the mean number of fiducials in the camera's field of view, also determines the probability of less than four fiducials being present in the field of view at an arbitrary time.

In Table II, P_0 denotes the probability of a *soft fiducial depletion failure*, which occurs when the tracking camera has less than four fiducials within its field of view. With only three fiducials in the tracking camera, there is the possibility of misconstruing the direction of an angular motion when crossing certain boundaries However brief failures of this sort will not usually lead to tracking errors. If an ambiguity in tracking were to occur during this period, it would be immediately resolved following the acquisition of a fourth fiducial.

P_1 denotes the probability of a *hard fiducial depletion failure*, which occurs when the tracking camera has less than three fiducials within its field of view. With only two fiducials in the tracking camera, it is impossible for the tracking algorithm to operate properly because the solution of six unknowns is impossible from the four numbers characterizing two data points Nevertheless, an algorithm can be instructed to perform tracking on the basis of an a priori model in which certain head motions (such as looking back and forth) are deemed more likely than others (such as stooping down with the head looking upwards, or standing more than six feet tall) Thus brief periods in such a state may not cause an objectionable tracking failure. Also, during such a state, the observation of the position of two fiducials will enable the tracker to, almost surely, know the identity of a third fiducial when it comes into view (however the tracker will never be sure of proper correspondence until four fiducials are in view).

P_2 denotes the probability of a *serious fiducial depletion failure* which occurs when the tracking camera has less than two fiducials within its field of view. With only one fiducial in the tracking camera, it is very difficult for the tracking algorithm to operate with any degree of accuracy

since the solution of six unknowns is utterly impossible from the two numbers characterizing one data point. Nevertheless, an algorithm can perform extremely crude tracking on the basis of a very oversimplified model of a priori head motion. Since one fiducial remains in the field of view, when additional fiducials are observed, the identification of those points will be simplified greatly by the continuing identification of the one point. However initial misidentification of a second and third fiducial may occur if significant changes in head orientation occur during the period of a serious fiducial depletion failure.

P_3 denotes the probability of a *catastrophic fiducial depletion failure*, which occurs when the tracking camera has absolutely no remaining fiducials within its field of view, at which point tracking stops. When a fiducial then re-enters the camera's field of view, the tracker may be able to make an educated guess as to its identity, but will generally have no confidence in that guess until the point at which four fiducials are again in view.

The calculation of these failure probabilities is based on a worst-case model in which the placement of fiducials is totally random, with the specified average density, so that the number of observed fiducials is governed by Phoisson statistics. In a real fiducial pattern, clumping of fiducials and areas of fiducial depletion will be avoided. Thus situations in which the expected number of fiducials in view N, exceeds 10, have zero probability of even a soft fiducial depletion failure, and the corresponding cells in Table II are left blank. For $6<N<10$, the numbers for the failure rates are somewhat pessimistic for the same reason. For $N<6$, the probability of fiducial depletion failures swells, making the task of reliable tracking tenuous (lines 7, 10, 15, 18) or impossible (lines 13, 16).

It should also be pointed out that the calculation of the expected number of fiducials in view N, is based on the camera axis being normal to the object plane: $\theta=\phi=0$. For the same z_o, an inclination of the camera's axis with respect to the board, will increase N, the expected number of fiducials within the field of view. For instance, using camera C in line 18 if the helmet were tilted by an angle of 30°, the expected number of fiducials in the camera's view N, would be doubled to 12.

In any case, however, a given camera field of view, and a given density of fiducials will define a minimum working distance z_{MIN}, below which fiducial depletion failures are likely to become troublesome. For a camera

with a horizontal acceptance angle θ_{FOV}, with a density of fiducials ρ, the expected number of fiducials in the camera's field of view will exceed five, as long as z_o exceeds z_{MIN} given by:

$$z_{MIN} = \frac{1.3}{\sqrt{\rho}\ \tan(\theta_{FOV}/2)}$$

In this equation, $1/\sqrt{\rho}$ can be viewed as the "average distance between fiducials," defined as the spacing between fiducials that would be obtained if they were arranged in a rectangular grid pattern. It can be seen that an increase in the camera field of view can dramatically decrease the number of fiducials required to be placed on the board. Attempting to accommodate small working distances, on the other hand, will rapidly drive up the number of required fiducials.

EXHIBIT C

TriSen Videometric Tracker

Design Review November 2, 1995

CONCLUSION

- The TriSen Videometric Tracker Will Work

WHAT NEXT?

- Complete Tracker Design
- Define Wiring Board System and Interfaces
- Prepare for February 1996 Demonstration

TriSen Videometric Tracker

Design Review November 2, 1995

Augmented Reality System

- **Configuration includes:**
  1. **Boeing Aircraft Cable Data Base & Interface**
  2. **Wiring Boards with Mapped Fiducials**
  3. **Cable Packs with Marked Wires Cut to Length**
  4. **Tracker System Including:**
     - **a. Head Mounted Display/Sensor**
     - **b. Body Worn Computer/Tracker Electronics**
     - **c. System Battery Pack**
     - **d. Battery Charger**
     - **e. System Carrying Case**

- **Boeing provides Controlled Storage for Tracker Systems & Battery Chargers**

Tracker Configuration 1

- **Head Mounted Display**

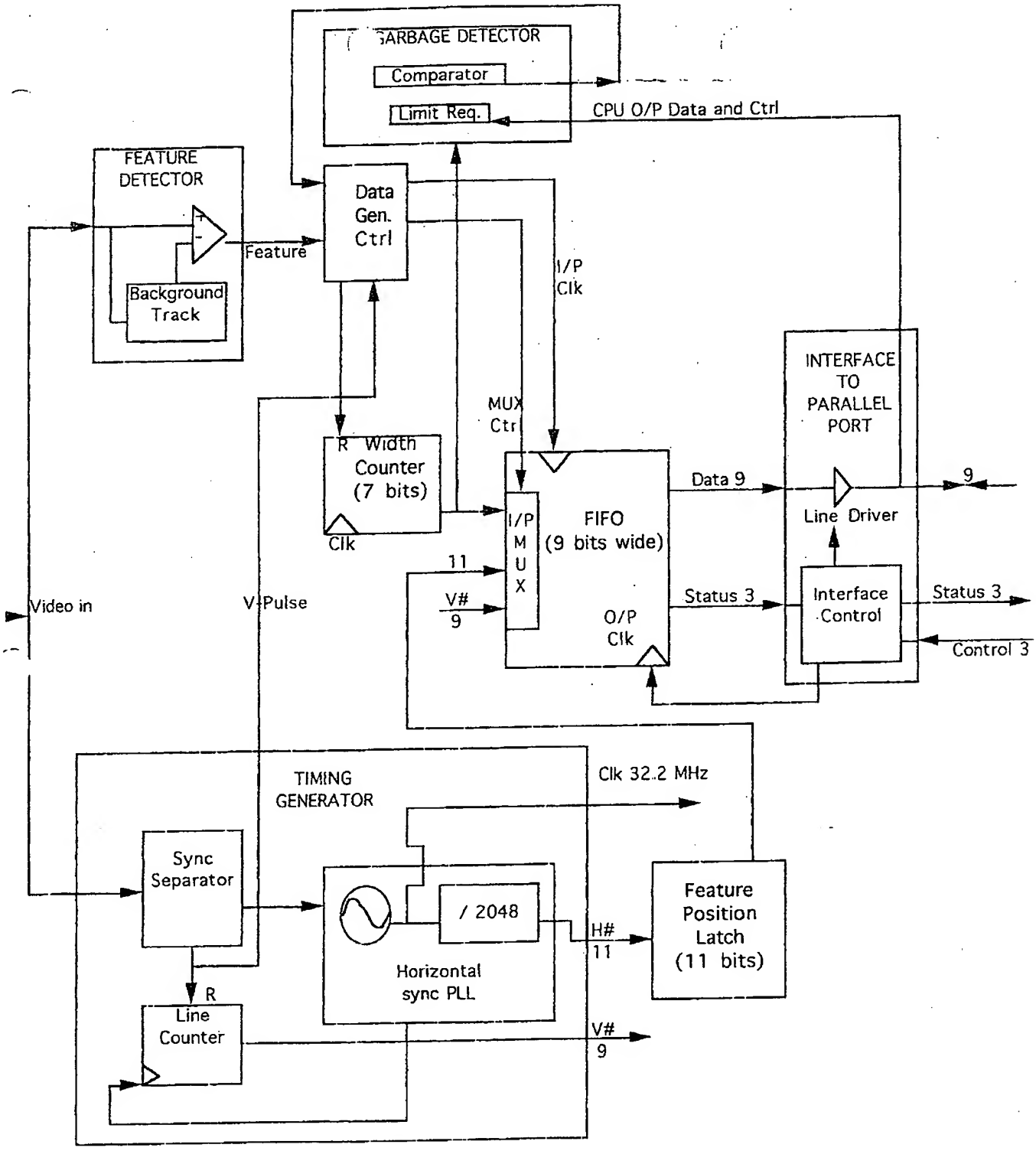
1. **Headband (Visionics)**
2. **Binocular "See Through Optics" Mounted on Headband Optical Bench**
3. **640 x 480 AMEL 0.7 Inch Diagonal Monochrome Image Sources, Including Drive Electronics with VGA Interface**
4. **Cables**
5. **TV Camera Mounted on Headband Optical Bench**

Tracker Configuration 2

- **Body Worn Computer**
  1. **80486 DX 75 Mhz, 8 MB RAM, 170 MB Hard Drive, 3 PCMCIA Slots, Enhanced Parallel Port (IEEE P1284-I) Communications Ports, Voice Recognition, Battery Pack**

- **Video Feature Extractor Card in PCMCIA Configuration (Includes 9V Power Supply for TV Camera)**

- **Cables, Interface**


GARBAGE DETECTOR
Comparator
Limit Reg.
CPU O/P Data and Ctrl
FEATURE DETECTOR
Background Track
Feature
Data Gen. Ctrl
I/P Clk
INTERFACE TO PARALLEL PORT
MUX Ctr
R Width Counter (7 bits)
Clk
I/P M U X
FIFO (9 bits wide)
O/P Clk
Data 9
Line Driver
9
11
V# 9
Status 3
Interface Control
Status 3
Control 3
Video in
V-Pulse
TIMING GENERATOR
Clk 32.2 MHz
Sync Separator
/ 2048
Horizontal sync PLL
H# 11
Feature Position Latch (11 bits)
R
Line Counter
V# 9

Video Feature Extractor

- **Breadboard Functioning Well**
- **Real Time Output at Field Rate (60Hz)**
- **Adaptive Contrast Enhancement**
- **Some Hardware "DeGarbage"**
  Output is "Runs"
- **392 Total Pins in Present Design, Connectors, DIPs, Analog Components**
- **Low Manaufacturing Cost Planned Design**
  PCMCIA Format

Tracker Demonstration

- **Video Feature Extractor**
  1. **Adaptive Contrast Enhancement**
  2. **Example Runs/Frame**
  3. **Hardware DeGarbage**

- **Clumping & Centroiding Examples**

- **Abberation Correction**

- **Ten Cycle Estimation Performance Position & Orientation**

- **Initial Correspondance**

Tracker System Software 1

1. VFE Input	40 Cm Z Difficult Case	I/O	2.6 msec
2. Clump	200 Runs		2.6 msec
3. Centroid	48 Fiducials		0.1 msec
4. Abberation Correction	48 Fiducials		0.5 msec

IF TRACKING

5. Tracking Correspondance & Estimation	9.0 msec
6. Calculate & Output Symbols to VGA Driver	1.0 msec

Tracker System Software 2

IF NOT TRACKING

• Initial Correspondance	900 msec
• Tracking Correspondance & Estimation	80 msec
• Calculate & Output Symbols to VGA Driver	1 msec

AGENDA

• Welcome	Jim Wolverton
• Purpose of Meeting	James Bath
• Augmented Reality System	Jim Wolverton
• Tracker Configuration	Jim Wolverton
• Video Feature Extractor	Jeffrey Meisner
• Tracker System Software	Jerry Wang
• Tracker Demonstration	JJJ

EXHIBIT D

To: mizell@espresso.rt.cs.boeing.com
cc: janin@espresso.rt.cs.b ing.com
Subj: Re: ARS Program
David:

(D)

Thanks for the note. Norm Tarleton is a nice guy, and intelligent, but he is powerless to help you.

We are working very, very hard to complete a production tracker demonstration for you mid next month. We will be able to produce the tracker at a "reasonable" price.

The VFE II is complete, tested, documented, and robust.

A copy of a status report I sent Jim Bath is appended below.

Thanks for your help.

Jim

Date: Tue, Mar 26, 1996 5:00 PM EDT
From: WolvertonJ
Subj: Confirmation of Work Direction
To: jbath@p04.mn17.honeywell.com
cc: lewandowski_ron@mn15-gw.mavd.honeywell.com, bender_barb@mn15-gw.mavd.honeywell.com

Ron & Barb please forward to James Bath. Your group has such an interesting email system that we may need to revert to snail mail and telephone tag to establish reliable communication. Enjoy the wonderful world of "Quick & Dirty Operating System" (DOS).

James:

Confirming our near term ARS tasks:

1. TriSen is completing integration of the "C" program on a 486DX 66 PC;

the program contains an assembly language module for control ofthe parallel port, (which allegedl, neets IEEE specification 128

The programming tasks include those for the Video Feature Extractor (VFE) Gate Array Logic chips; the chips provide the necessary flexibility for this stage of development. We have completed and tested the programming for VFE II. We have excellent documentation for that programming.

We have written a draft Applications Program Interface (API) for operation with the Boeing software. Adam Janin and we believe it will work, but the proof will be demonstrated in Seattle.

We have a well documented Forth language implementation of initial correspondance. We will transfer this implementation to "C" after completion of the higher priority programing.

We have completed a significant testing regime and are completely rewriting the "C" implementation of the clumping, centroiding, lens aberration correction, and tracking modules. These are being documented as they are being written with the participation of our "C" programmer.

(2) We, and you, are awaiting the delivery of the backpack 486 computer with its new parallel port. We are hoping the "C" program integrates easily in that computer processor, but are concerned about the new parallel port.

We are also concerned about the "real" battery voltage available. We will need to convert it, using a power supply on the VFE, to drive the head mounted TV camera and the VFE electronics. Depending upon the value and "droop" of the voltage, this integration may be easy or require a complete redesign of that portion of the VFE.

(3) We are also preparing to integrate with the Flex "wearable computer." The same risks apply.

(4) We have designed, and have in fabrication, the production prototype wiring boards. They will be robust, stable, mechanically sound, and

provide both "white dot" and "black dot" instantiations.

We will need to determine the lighting conditions at Boeing, user color preferences, etc. to choose final colors and lighting supplement (if required).

(5) We have completed and checked out two ("white dot" and "black dot") instantiations of VFE II. The final schematics are being entered into the computer to generate "good copies" and net lists.

Have a fun telecon with Boeing tomorrow.

Jim

EXHIBIT E

Communications between the VFE and a P.C.

(E

Following is a description of communication protocol applicable to the Video Feature Extractor, version 2, using a P.C. with an Enhanced Parallel Port (EPP) interface. This description applies to the circuitry as modified according to changes specified on March 11 and March 14, 1996. Also note that the "clear-to-end-of-field" command has been eliminated as of the 7-9-96 version of U7.

Control of the VFE to PC interface is performed by U7 which also controls output from the FIFO's (U4 and U5), and U6 to the parallel port, and which additionally directs input to the VFE from the parallel port to U6 and U3. The FIFO's (U4 and U5) have all 8 data bits hooked up to the bi-directional port in the computer, as does U6. U3 receives input only from data bits 1-5 of the parallel port, and the control chip, U7, is hooked up to data bit 1 of the port, in addition to several control signals.

There are 4 control signals from the computer feeding U7, and U7 sends two control signals to the computer (in addition to the hookup to data bit #1). The EPP control signals driving U7 are:

1) /Datastrobe or /DS4 from EPP pin 14 driving U7 pin 9.

2) /Write or /W from EPP pin 1 driving U7 pin 10.

3) /Addrstrobe or /DS3 from EPP pin 17 driving U7 pin 15 (as per the March 11, 1996 modifications).

4) /RESET or ALT from EPP pin 16 driving U7 pin 8.

The VFE sends to the computer three control signals:

1) /WAIT from U7 pin 23 to EPP pin 11.

2) ERROR from U7 pin 22 to EPP pin 15.

3) /INTERRUPT from U6 pin 22 to EPP pin 10.

Of these signals, /DS4, /DS3, /W, and /WAIT, are interlocked handshaking signals involved in EPP reads or writes at device addresses PORT+4 or PORT+3. /ALT is controlled by writing bit 3 (04h) of device address PORT+2. ERROR is read as bit 4 (08h) of device address PORT+1. /INTERRUPT causes a parallel port interrupt (usually IRQ7, interrupt

vector 0Fh) if interrupts have been enabled by setting bit 5 (10h) of device address PORT+2.

Internally to the VFE, data to or from the EPP are controlled by the following control signals generated by U7. Output from U6 to the 8 data bits of the parallel port are controlled by the signals OC1 and OC2 generated by U7. The **interrupt mode register** (internal to U6) is loaded by U7 clocking the signal **INTERRUPT-SET**. Outputs from the FIFO's, U5 and U4, to the parallel port data lines are enabled by assertion of **/READ-1** and **/READ-2** respectively (the falling edge of these signals also clock out the data just read). Data from parallel port data lines 1-5 are latched into the **garbage limit register** (internal to U3) by U7 clocking the signal **/GARBAGE-SET**. Finally, all data is cleared from both FIFO's, and data acquisition by U3 is halted (until the beginning of the next video field) by U7 asserting **/FIFO-RESET**.

The signals **OC2** and **OC1** determine the output state of U6 as follows:

- If OC2=0 and OC1=0, U6 does not drive the parallel port data lines.
- If OC2=0 and OC1=1, U6 drives the parallel port data lines with all zeros (00h).
- If OC2=1 and OC1=1, U6 drives the parallel port data lines with the current time.
- If OC2=1 and OC1=0, U6 drives the parallel port data lines 3-8 with zeros, drives bit 2 with status information (with a 1 indicating that the FIFO's are at least half-full), and does not drive bit 1, allowing it to be driven by U7.

Programming Procedures

Control of the above functions by the CPU is through byte or word INPUT and OUTPUT instructions at device addresses PORT through PORT+4, where PORT is the base address of the enhanced parallel port adapter, typically 378h. Following are valid output commands:

• Outputting a 4 to PORT+2 will set the ALT bit detected by U7 pin 8. Outputting a 0 to PORT+2 w[illegible] clear the ALT bit. The [illegible]tus of the ALT bit will affect the meaning of other input and output operations to the VFE!

• Outputting a byte to PORT+4 will initiate a write cycle in which W and DS4 are asserted. U7 responds by clearing WAIT after which the write cycle ends. If ALT=0, U7 clocks the INTERRUPT-SET signal so that the interrupt register in U6 is loaded from the parallel port data lines. However if ALT=1, then U7 performs a master reset by pulsing FIFO-RESET.

• Outputting a byte to PORT+3 will initiate a write cycle in which W and DS3 are asserted. U7 responds by clearing WAIT after which the write cycle ends. If ALT=0, U7 clocks the GARBAGE-SET signal so that the garbage register in U3 is loaded from parallel port data lines 1-5.

Following are valid input operations:

• 16 bit data is read from the FIFO's by setting ALT=0 and performing a word input from PORT+3 which initiates a double read cycle which occurs in the following manner. First the EPP adapter asserts DS4 in response to which U7 sends out READ1 which causes FIFO #1 to present data onto the parallel port data lines (this is the high byte of the data word). U7 clears WAIT in response to which the EPP adapter clears DS4. Then the EPP adapter asserts DS3. U7 then pulses READ2 causing FIFO #2 to drive the data lines with the low byte of the data word. U7 clears WAIT in response to which the EPP adapter ends the entire input cycle and sends the 16 bit data to the CPU.

The format for this data word is described in a separate document. In the case that an input operation is initiated when there is no data present in the FIFO, then U7, rather than enabling output from the FIFO's, instructs U6 to drive all 8 parallel port lines with zeros for each byte transfer. Thus an input data word of 0 signifies an empty FIFO (there is no possible data word which will be exactly 0).

While performing a word input from PORT+3 is the recommended form of inputting data from the FIFO's, byte inputs from PORT+3 or PORT+4, or word inputs from PORT+4, (with ALT=0) will cause some of the same strobes to clock data out of the FIFO's. For instance, a 16 bit input from PORT+4 will cause two inputs to occur via the DS4 strobe, which will confuse the VFE. The VFE expects an input cycle via DS4 to be followed by an input cycle via DS3. So a word input from PORT+4 will

set an error condition in the VFE, as will most other combinations of byte inputs when ALT=1.

• With ALT=1, inputting a byte from PORT+3 initiates a read cycle in which DS3 is asserted. In this case the VFE status is read, in which U7 drives data line 1 with status information and U6 is instructed (by OC2=1, OC1=0) to drive data line 2 with status information. Data lines 3-8 are each driven zero by U6. The resulting status code is interpreted as follows:

0 No data present in FIFO.

1 Data ready in FIFO.

2 FIFO is full and has overflowed. If no data reads have occurred since the onset of the overflow, then up to 1024 valid data words may be read, if desired. Following that (or instead of that) a master reset is strongly recommended in order to prevent skewed data or video fields with missing runs. Note that if a FIFO overflow occurs followed by one or more FIFO read operations so that the FIFO is no longer full, then this condition will no longer be read. However the error condition (see below) will be set until a master reset is performed.

3 FIFO is at least half full (512 data words present, for a VFE containing the CY7C425 chips in the U4 and U5 sockets).

• Inputting a byte from PORT+4 in the case that ALT=1, causes a read cycle in which the EPP port asserts DS4. This causes the VFE time to be read, in which U6 is instructed (by OC2=1, OC1=1) to drive the 8 data lines with the 8 most significant bits of the VFE time base.

• Inputting from PORT+1 is used for reading the ERROR bit from pin 22 of U7. Its value is presented as bit 4 (08h) of the input data. The 08h bit indicates an error condition. An error condition signifies the detection of a FIFO data skew or a FIFO overflow. The error condition must be cleared by performing a master reset, as described above. However if the error is a result of a FIFO overflow, it is possible to first read out any valid data residing in the FIFO.

These input and output operations are summarized in the following table.

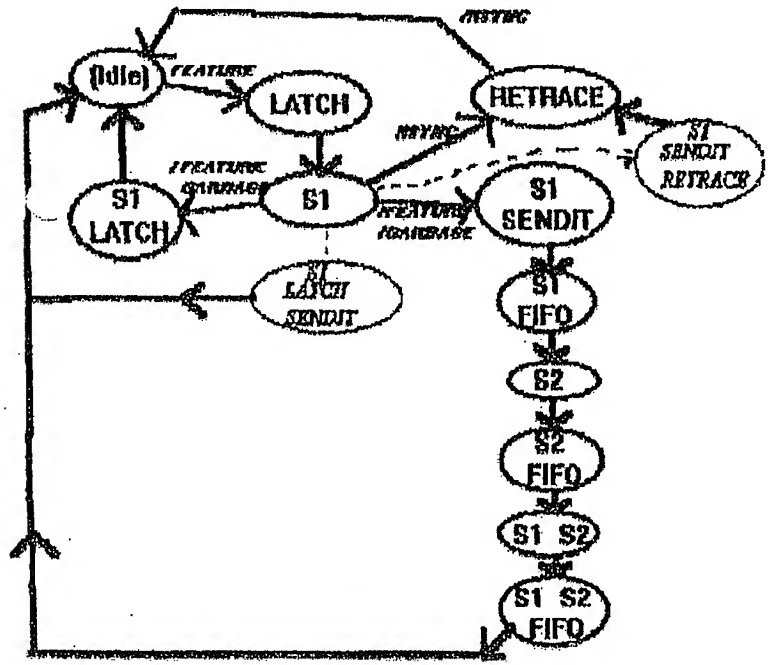
R/W	PORT	Byte or Word	ALT bit	Operation	Data format
WRITE	PORT+2	x	x	Set/reset ALT	xxxx xAxx
WRITE	PORT+4	BYTE	0	Interrupt Set	IIII IIII
WRITE	PORT+4	x	1	Master Reset	xxxx xxxx
WRITE	PORT+3	BYTE	0	Garbage Set	xxxG GGGG
READ	PORT+3	WORD	0	Read FIFO data	(see text)
READ	PORT+3	BYTE	1	Read STATUS	0000 00SS
READ	PORT+4	BYTE	1	Read TIME	TTTT TTTT
READ	PORT+1	x	x	Read ERROR	xxxx Exxx

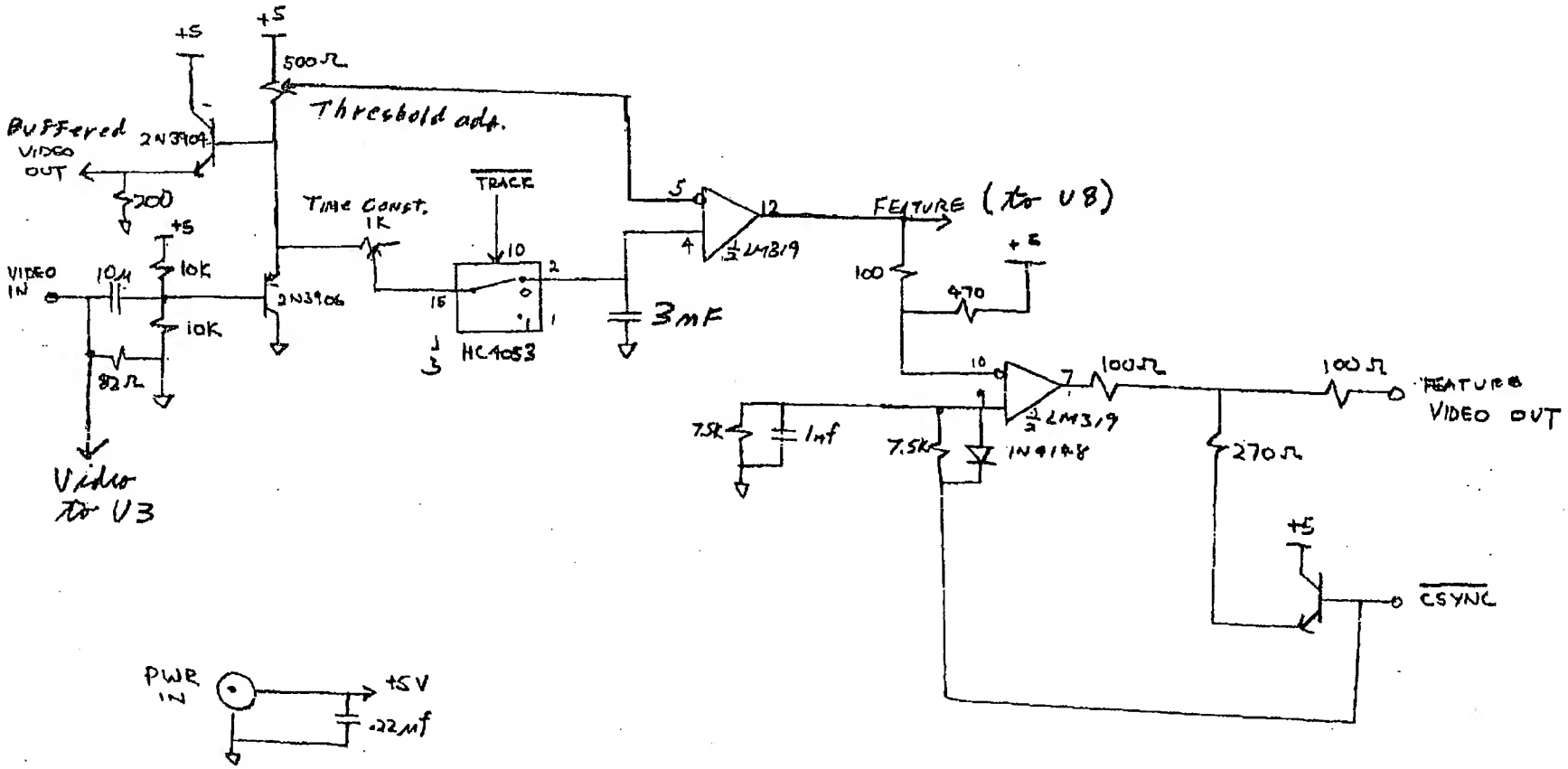
(x="don't care")

EXHIBIT F

(4)

DATACTRL U8 State Diagram





Feature Generator

VFE p1/7

10-30-95

EQUATIONS

```
/H1 := /H1*S1*/S2 + V2*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH1V2 :=  [OE=S2]  [E=S1*/S2 + /S1*/S2 +S1*S2] [C=CLK]  ;
H2 := /H2*S1*/S2 + V3*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH2V3 :=  [OE=S2]  [E=S1*/S2*H1 + /S1*/S2 +S1*S2] [C=CLK] ;
   := /H3*S1*/S2 + V4*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH3V4 :=  [OE=S2]  [E=S1*/S2*H1*H2 + /S1*/S2 +S1*S2] [C=CLK] ;
H4 := /H4*S1*/S2 + V5*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH4V5 :=  [OE=S2]  [E=S1*/S2*H1*H2*H3 + /S1*/S2 +S1*S2] [C=CLK] ;
H5 := /H5*S1*/S2 + V6*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH5V6 :=  [OE=S2]  [E=S1*/S2*H1*H2*H3*H4 + /S1*/S2 +S1*S2] [C=CLK] ;
H6 := /H6*S1*/S2 + V7*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH6V7 :=  [OE=S2]  [E=S1*/S2*H1*H2*H3*H4*H5 + /S1*/S2 +S1*S2] [C=CLK]
H7 := /H7*S1*/S2 + V8*S1*S2 + S1*/S2*H1*H2*H3*H4*H5*H6*H7 ;
          /xH7V8 :=  [OE=S2]  [E=S1*/S2*H1*H2*H3*H4*H5*H6 + /S1*/S2 +S1*S2] [C=CL

xV1 := /V1 [C=/H11 ] [OE=S1*S2] ;
V2 := /V2  [C=/V1] ;
V3 := /V3  [C=/V2] ;
V4 := /V4  [C=/V3] ;
V5 := /V5  [C=/V4] ;
V6 := /V6  [C=/V5] ;
V7 := /V7  [C=/V6] ;
V8 := /V8  [C=/V7] ;

RESET = VPULSE  ;     Just in order to reset the V counter!!


G1 :=  [OE=OFF] [C=CLK]  ;   Configuration equation really for OVERSIZE

UNDERSIZE := UNDERSIZE*(G4*/H5 + G3*/H4 + G2*/H3 + G1*/H2 ) + /S1  [C=CLK]
/OVERSIZE := /OVERSIZE*(G8*/H7 + G7*/H6 + G6*/H5 + G5*/H4 )*/VPULSE + /S1
                + /G8*/G7*/G6*/G5 ;  <<< to disable checking with 0 input

GARBAGE = OVERSIZE + UNDERSIZE +/V8*/V7*/V6*/V5*/V4  ;
```

TITLE: U13 - Output control chip for CPU interface to VME

CPU Output to register 2:

Reg 2 value (hex)	Strobe2	Strobe1	effect
0	0	0	idle
1	0	1	Sets garbage limits
2	1	0	Readout FIFO data
3	1	1	Master Resett (Discard FIFO data)
10 (+ above)			Enable Parallel port interrupt

CPU Read from register 1 --- DIFFERENT FOR VERSION 2!!

Reg 1 value as read (hex)	Status3	Status2	Status1	Meaning
0	0	0	0	idle
10	0	0	1	DATA READY (After H Sync pu
20	0	1	0	Data OVERFLOW
30	0	1	1	FIFO is HALF FULL
40 (+above)	1			VPULSE inactive
00 (+above)	0			VPULSE active

(also pulses low when HALF FULL rises)
(If enabled, hi-lo transition on this pin causes an interrupt through parallel port, IRQ7, when using LP

VERSION # 4 DATE 10-27-95

```
DEVICE GAL16V8
ES     'Output'
OPERATORS  * + $ /

PINS
CLK, OE-GND    1,11    ;
/HSYNC, /HF, DR, /OVFLW, /VPULSE, /PWRUP  2,3,4,5,6,7  ;   INPUTS
/STROBE1 8 ;  Input from par. port pin 14
/STROBE2 9 ;  Input from par. port pin 1

STATUS1 15 ; Output to par. port pin 13
STATUS2 14 ; Output to par. port pin 12
STATUS3 16 ; Output to par. port pin 10
```

```
/READOUT 17   ; Clock output of FIFO
/CPU-COMMAND 18 ;
/MR       19 ;
GOTSTROBE 13 ;  Internal state for debouncing input strobes


EQUATIONS

GOTSTROBE :=  STROBE1 + STROBE2 ;

CPU-COMMAND := /READOUT*/CPU-COMMAND*/MR*GOTSTROBE*/STROBE2*STROBE1
               + CPU-COMMAND*GOTSTROBE ;

READOUT := /READOUT*/CPU-COMMAND*/MR*GOTSTROBE*STROBE2*/STROBE1
         + READOUT*GOTSTROBE ;

MR := /CPU-COMMAND*/MR*GOTSTROBE*STROBE2*STROBE1 + MR*GOTSTROBE
      + MR*/VPULSE       + PWRUP  ;


STATUS1 := DR*/OVFLW*STATUS1 + DR*/OVFLW*HSYNC + HF*/OVFLW ;
STATUS2 := HF*/STATUS3 + HF*STATUS2 + OVFLW ;

/STATUS3 := VPULSE + HF*/STATUS2 ;
```

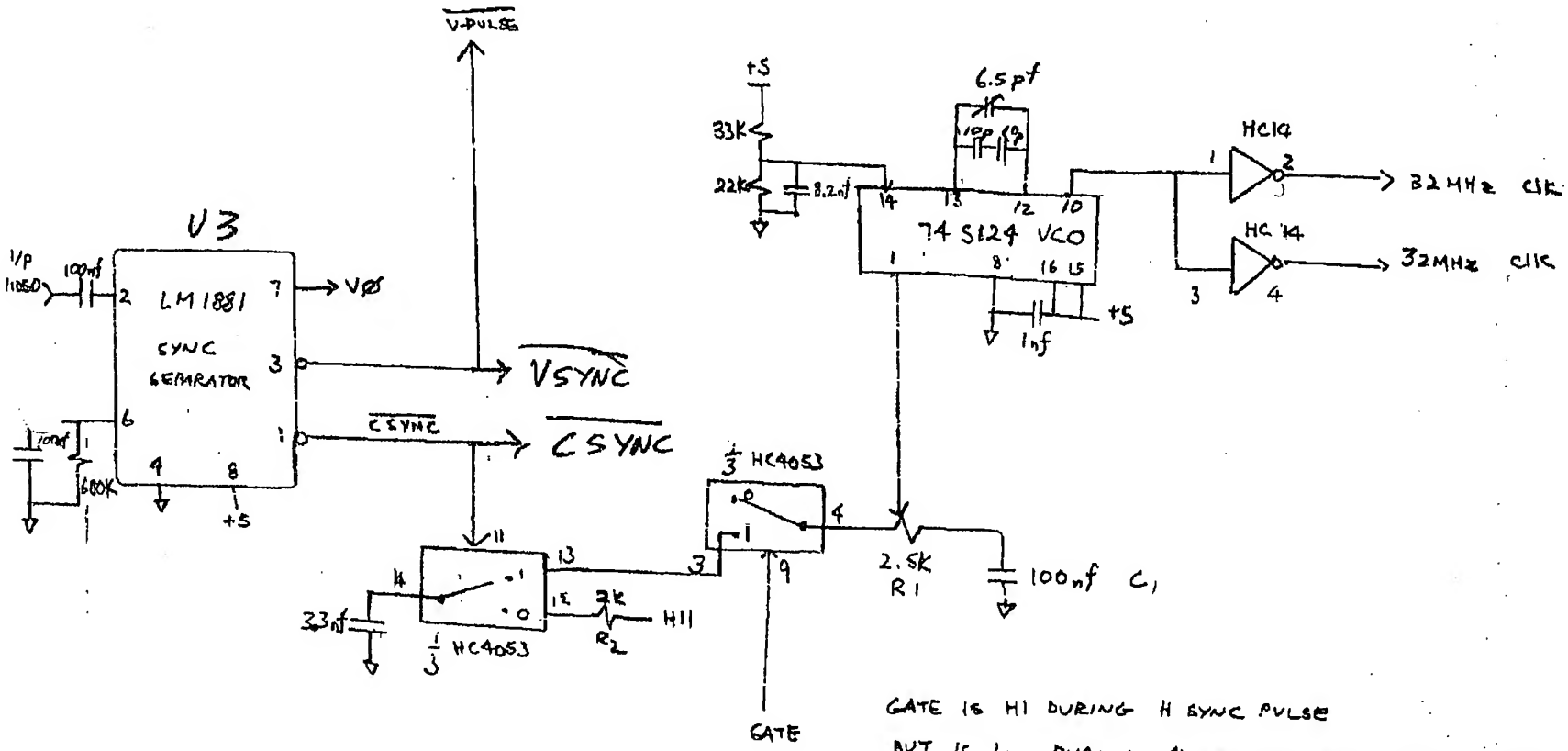
-------- C:\PLAQ\TRISEN\VFE\CNTLATCH\DATAFRMT.TXT

OUTPUT DATA ORGANIZATION - VERSION 2 (10-16-95)

Order of presentation	Select lines S2	Select lines S1	Data organization - FIFO data lines D9	D8	D7	D6	D5	D4	D3	D2	D1
1	0	1	P11	P10	P9	P8	P7	P6	P5	P4	P3
2	1	0	W7	W6	W5	W4	W3	W2	W1	P2	P1
3	1	1	V8	V7	V6	V5	V4	V3	V2	V1	V0
Parallel port data line #			8	7	6	5	4	3	2	1	*
Parallel port DB-25 pin #			9	8	7	6	5	4	3	2	11

* Read on status port, bit 80h

P(11-1) = Position of beginning of run
W(7-1) = Width of run
V(8-0) = Vertical line of run



PLL DESIGN FORMULA

LOOP POLE FREQ

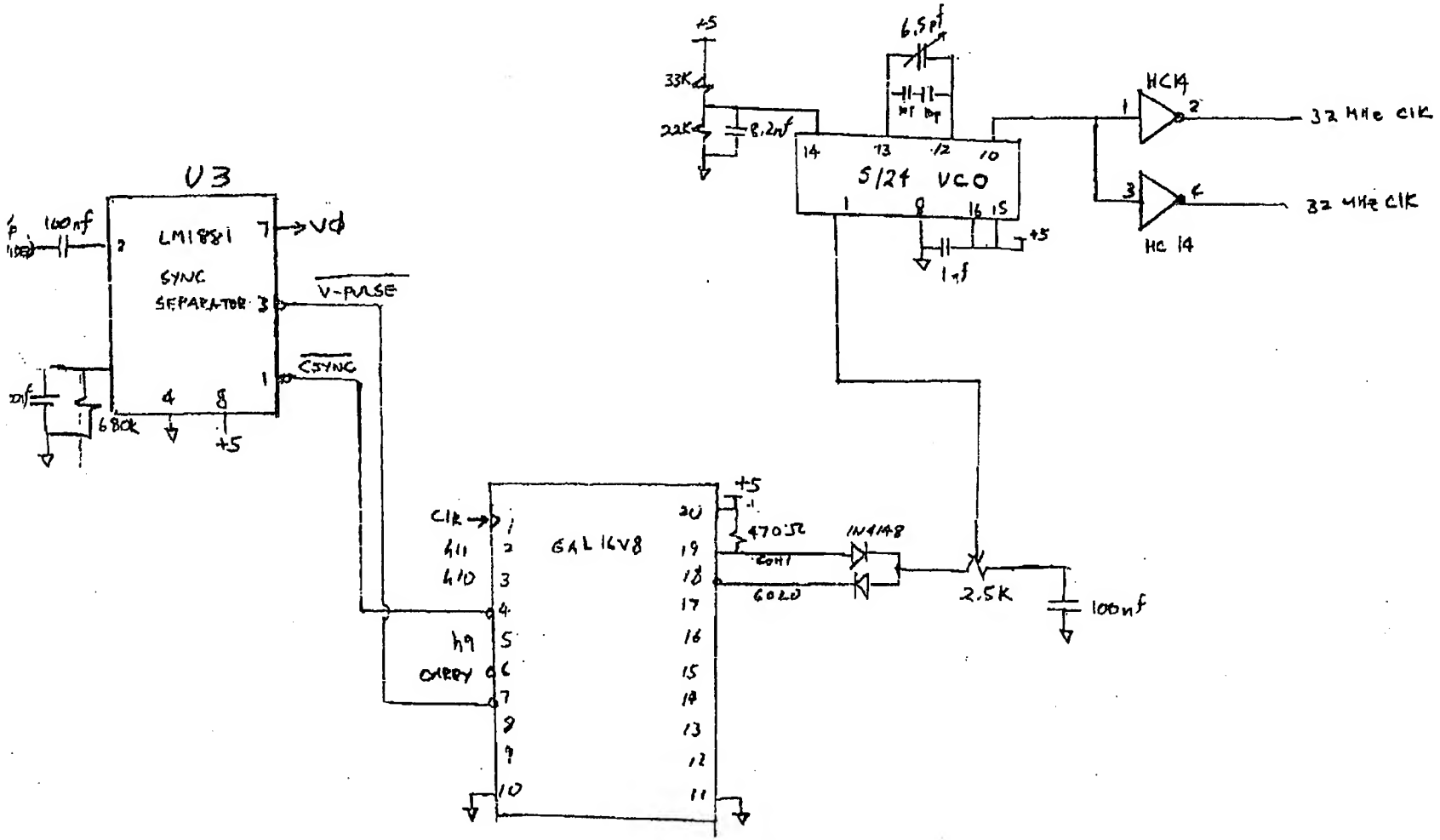
$$\omega_0 = \sqrt{\frac{(.2)(15750\ \text{sec}^{-1})}{2\, C_1 R_2}}$$

Q (ADJUSTABLE) MINIMUM @

$$Q_{min} = \sqrt{\frac{2 R_2/R_1}{(.2) R_1 C_1 15750\ \text{sec}^{-1}}}$$

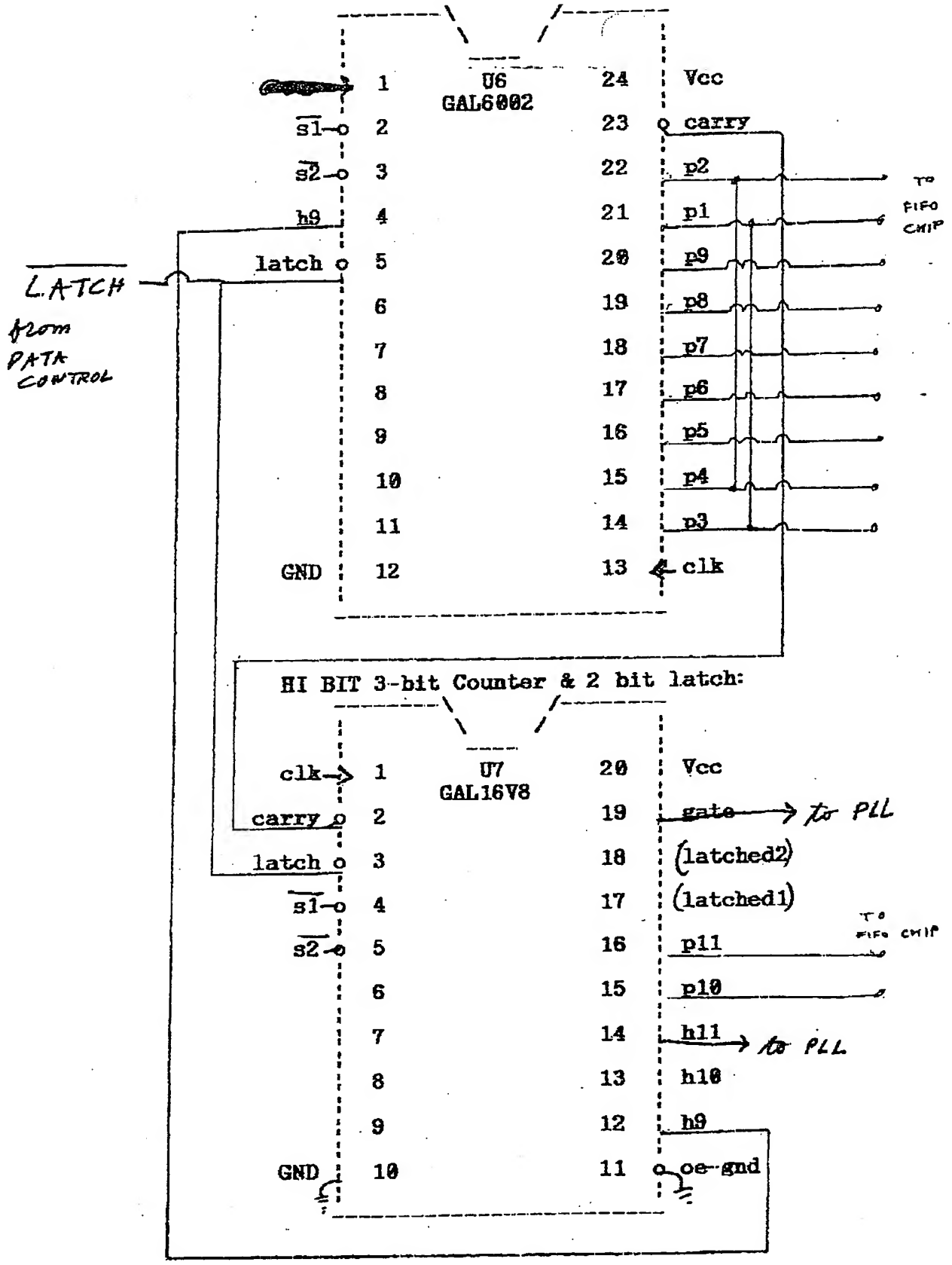
PLL

VFE p 217


U3
LM1881
SYNC
SEPARATOR
100nf
68K
+5
V-PULSE
CSYNC
GAL16V8
CLK
h11
h10
h9
CARRY
+5
470Ω
1N4148
COLD
2.5K
100nf
6.5pf
33K
22K
8.2nf
5/24 VCO
1nf
+5
HCU
HC 14
32 MHz CLK
32 MHz CLK

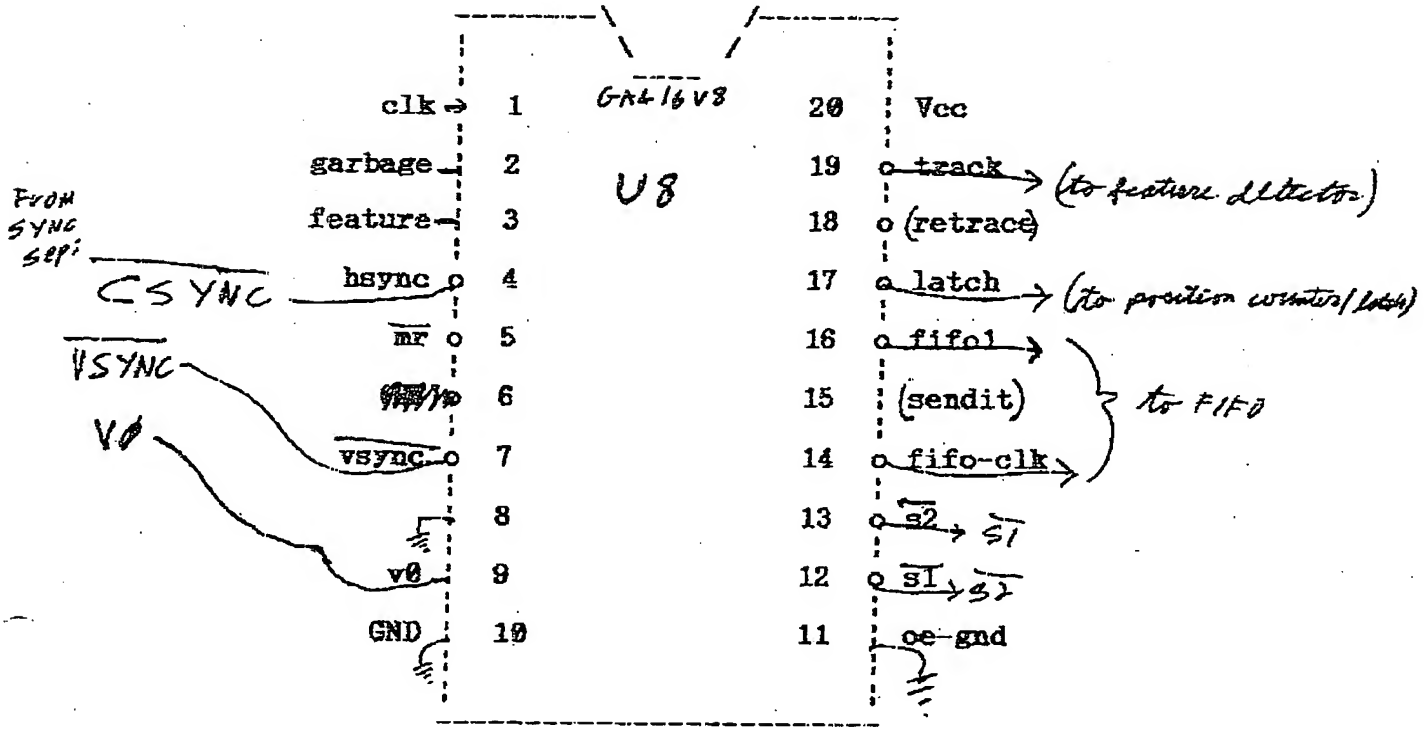
PLL

VFE p317



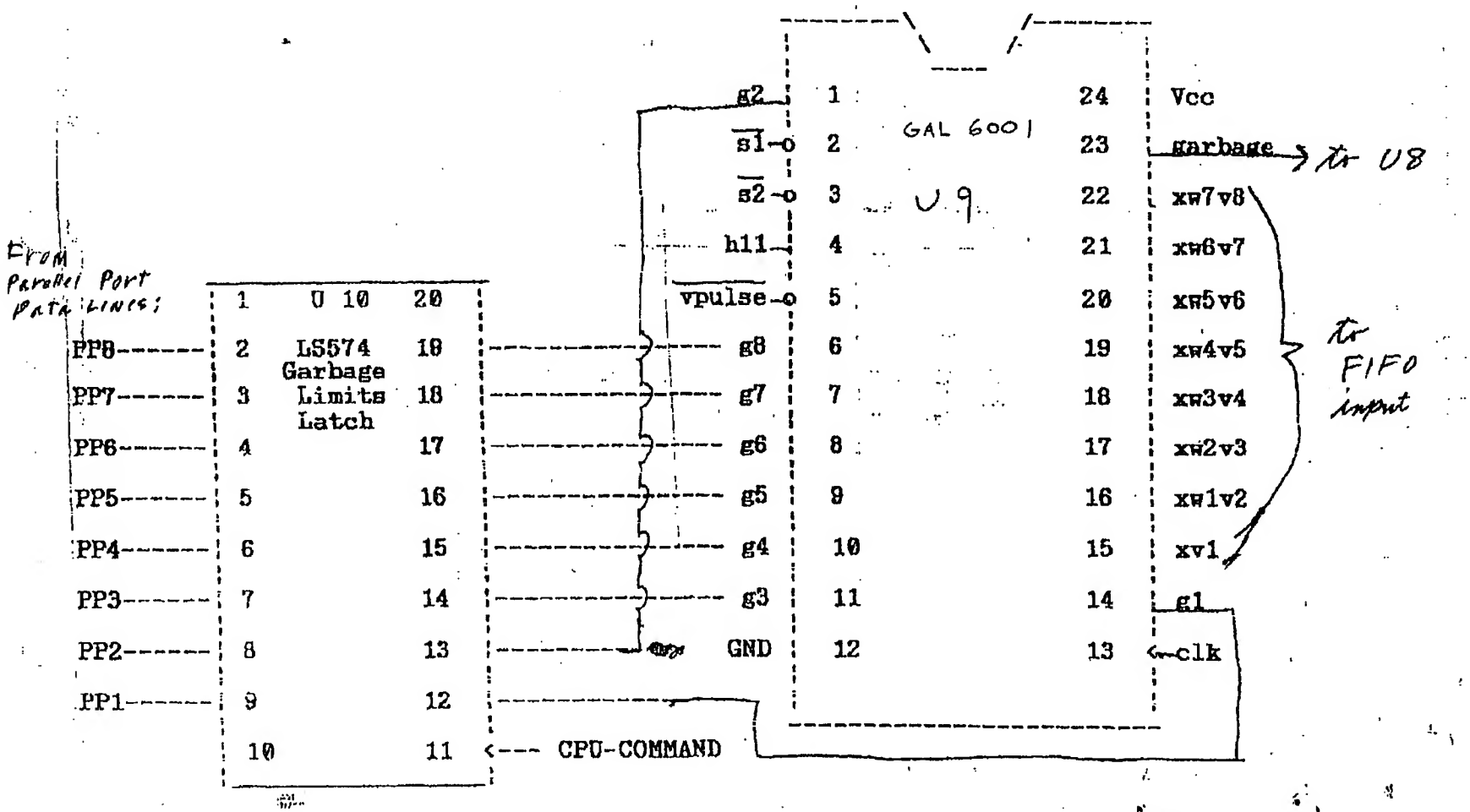
11-bit H counter VFE

TITLE: Data Generation Control chip for VFE



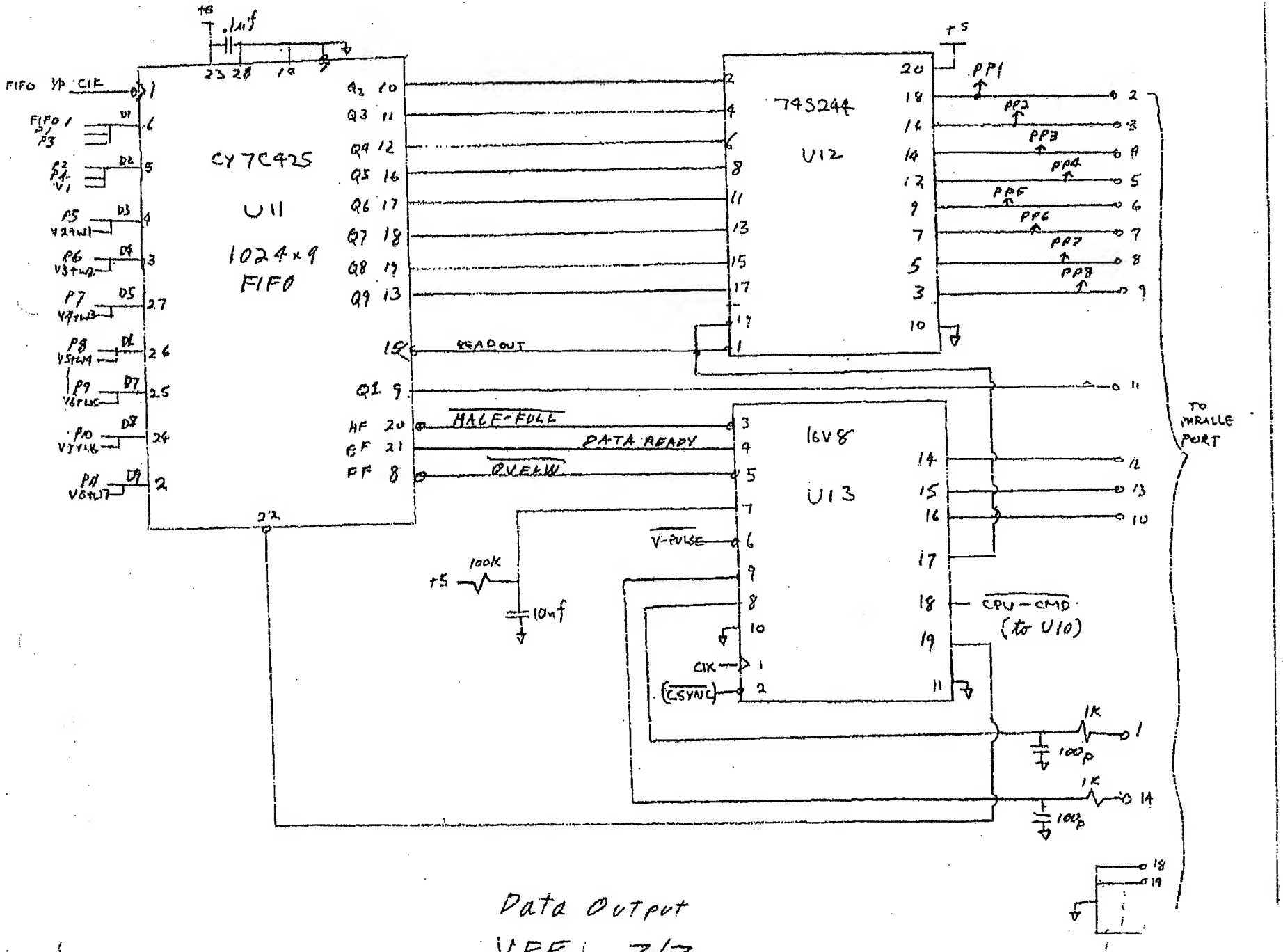
Data Control

VFE p5/7



Vertical Counter,
Width Counter,
& Garbage detector

VFE p\ 7


CY7C425
U11
1024x9
FIFO
74S244
U12
16V8
U13
READOUT
HALF-FULL
DATA READY
OVERW
V-PULSE
CPU-CMD
(to U10)
CLK
CSYNC
100K
10nf
1K
100p
TO PARALLEL PORT
Data Output
VFE 7/7

-------- C:\PLAQ\TRISEN\V~~\CNTLATCH\HICNT3.PLQ

TITLE: U7 - HI BIT 3-bit Counter & 2 bit latch for VFE horizontal clock
RECOVERY

THIS IS the highest order chip, counts 3 bits but only latches 2 bits
and implements other misc. functions
THIS IS FOR THE VERSION #2 CIRCUIT 9-6-95

VERSION # 5 DATE 10-16-95

```
DEVICE GAL16V8
ES    'hicnt'
OPERATORS  * + $ /

PINS
CLK, OE-GND    1,11    ;
/carry, /latch, /s1, /s2  2,3,4,5 ;   INPUTS
h9, h10, h11,  latched1, latched2, p10, p11, gate   12,13,14,17,18,15,16,19
 OUTPUTS
EQUATIONS

h9 := h9*/carry + /h9*carry ;
h10 := h10*/carry + /h10*carry*h9 + h10*/h9 ;
h11 := h11*/carry + /h11*carry*h9*h10 + h11*/h9 + h11*/h10 ;

latched1 := latched1*/latch + h10*latch ;
latched2 := latched2*/latch + h11*latch ;

p10=latched1   [OE=s1*/s2]  ;
p11=latched2   [OE=s1*/s2]  ;

gate := gate*/carry + gate*/h9 + gate*h11 + h9*h11*carry ;
```

```
-------- C:\PLAQ\TRISF~1\VFE\CNTLATCH\CNT8.PLQ

TITLE: 8 bit counter with 9 bit latch for 11 bit horizontal counter, VFE U6

VERSION 3 for the NEW DATA CONFIGURATION

Version 2 corrects the polarity of the p1 line (fifo q1)

VERSION # 3    DATE 10-11-95


DEVICE GAL39V18
ES     'cnt8latch'
OPERATORS  * + $ /

PINS

CLK     13     ;

/S1, /S2, H9, /LATCH  2,3,4,5  ;    INPUTS

  /P1,P2,/P3,P4,P5,P6,P7,P8,P9,/CARRY   21,22,14,15,16,17,18,19,20,23  ;
;  /1  2  /1  2  3  4  5  6  7
;  ^^^^Wired to fifo bits^^^^^

H1,H2,H3,H4,H5,H6,H7,H8  25,26,27,28,29,30,31,32  ;  Buried counter bits


EQUATIONS

H1 := /H1 [C=CLK] ;
H2 := /H2 [C=CLK] [E=H1] ;
H3 := /H3 [C=CLK] [E=H1*H2] ;
H4 := /H4 [C=CLK] [E=H1*H2*H3] ;
H5 := /H5 [C=CLK] [E=H1*H2*H3*H4] ;
H6 := /H6 [C=CLK] [E=H1*H2*H3*H4*H5] ;
H7 := /H7 [C=CLK] [E=H1*H2*H3*H4*H5*H6] ;
H8 := /H8 [C=CLK] [E=H1*H2*H3*H4*H5*H6*H7] ;

P1 := H1 [C=CLK] [E=LATCH] [OE=/S1*S2] ;
P2 := H2 [C=CLK] [E=LATCH] [OE=/S1*S2] ;
P3 := H3 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P4 := H4 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P5 := H5 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P6 := H6 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P7 := H7 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P8 := H8 [C=CLK] [E=LATCH] [OE=S1*/S2] ;
P9 := H9 [C=CLK] [E=LATCH] [OE=S1*/S2] ;

CARRY := /H1*H2*H3*H4*H5*H6*H7*H8 [C=CLK] ;
```

```
--------- C:\PLAQ\TRIS   \VFE\DATACTRL\DATACTRL.PLC

TITLE:  U8 - Data Generation Control chip for VFE

Version 6 sends out the TRACK signal to do background tracking when
          /feature, and right AFTER the hsync signal as long as feature
          is still on

Version 5 for NEW DATA CONFIGURATION will drive the V0 signal
         onto the FIFO1 line

Sends out signals: S1, S2, and FIFO-CLK for inputting data
into FIFO


VERSION # 6    DATE 10-13-95


DEVICE GAL16V8
ES     'datactrl'
OPERATORS  * + $ /

PINS
CLK, OE-GND    1,11    ; Ground the /OE pin, if not using to disable outputs
GARBAGE, FEATURE, /HSYNC   2,3,4  ;   INPUTS

/MR  5 ;          /HF 6 ; <<<< Not used in this version
/VSYNC  7 ; vertical sync input
V0 9 ; input of odd/even field from sync separator
/FIFO1 16 ; output of V0 when S1*S2
/S1, /S2, /FIFO-CLK, SENDIT,  /LATCH, /RETRACE  12,13,14,15,17,18 ;
/TRACK 19 ;  To cause tracking of background while active

EQUATIONS

LATCH := FEATURE*/S1*/S2*/RETRACE*/LATCH
      + S1*/S2*/SENDIT*/FIFO-CLK*/FEATURE*GARBAGE*/LATCH ; << Comes on to
                                                          ; INVALIDATE garbage data
RETRACE := HSYNC*/FIFO-CLK*/S2*/SENDIT  + RETRACE*FEATURE + RETRACE*VSYNC ;

TRACK = /FEATURE + RETRACE*/HSYNC*/VSYNC ;

S1 :=   LATCH*/S1*/RETRACE + S1*/S2*/FIFO-CLK*/LATCH*/RETRACE
        + S2*FIFO-CLK*/S1 + S1*S2*/FIFO-CLK ;

SENDIT := S1*/S2*/FIFO-CLK*/FEATURE*/GARBAGE*/LATCH*/SENDIT ;

S2 := S1*FIFO-CLK*/S2 + S2*/FIFO-CLK  + S2*/S1 ;

FIFO-CLK := S1*/FIFO-CLK*SENDIT*/LATCH + S2*/FIFO-CLK ;

FIFO1 = V0 [OE=S1*S2]  ;
```

-------- C:\PLAQ\TRIS[illegible]\VFE\PHASEDET\PHASEDET.PLC

```
TITLE: Phase detector, version 2, for VFE

Will someday become integrated with the horiz counter/latch for
better utilization.

Has separate GOHI and GOLO outputs, to be
connected to CV filter thru DIODES!


VERSION # 2     DATE 10-26-95


DEVICE GAL16V8
ES     'PhaseDet'
OPERATORS  * + $ /

PINS
CLK, OE-GND     1,11     ; Ground the /OE pin, if not using to disable
H10, H11, /CSYNC, H9, /CARRY  3,2,4,5,6 ;                    OUTPUTS
GOHI, /GOLO, CSYNC2, CSYNC3  19,18,12,13 ;

EQUATIONS

CSYNC2 := CSYNC ;  Latch in CSYNC
CSYNC3 := CSYNC2 ;  Delay CSYNC2 by one clock cycle


GOHI := H11*H10*/CSYNC3*CSYNC2*/CARRY + GOHI*H11*H10*/GOLO ;

GOLO := H11*H10*H9*CARRY*/GOHI*(/CSYNC2+CSYNC3) + GOLO*/CSYNC2
      +GOLO*CSYNC3 ;
```

```
New VERSION 4 : Will limi  width counter at 127 & st  counting;
                also sett g garbage G4-7 at 0 will disable oversize rejection
                                                          REJECTION
NO LONGER : Will stop the vertical counter at 255 (line# 510 or 511)
Rather Version 7 : flags all runs as GARBAGE after the vertical counter
      has wrapped around, and is less than 8

UNDERSIZE is RESETT when W4,W3.W2,W1 = G4,G3,G2,G1
OVERSIZE  is SETT   when W7,W6,W5,W4 = G7,G6,G5,G4

New pinout with version 3
New pinout with version 4

VERSION # 7     DATE 10-19-95


DEVICE GAL39V18
ES     'multi-cnt'
OPERATORS  * + $ /

PINS

CLK     13     ;

G8,G7,G6,G5,G4,G3,G2, /S1, /S2, H11, /VPULSE  6,7,8,9,10,11,1,2,3,4,5; INPUTS

; input on pin 14, using OLMC for state FF :
OVERSIZE 34 ;  state FF
G1 14 ;        input

  outputs:
GARBAGE 23 ;
xV1,xW1V2,xW2V3,xW3V4,xW4V5,xW5V6,xW6V7,xW7V8  15,16,17,18,19,20,21,22; PHYSICA
/V1,/W1,/W2,/W3,/W4,/W5,/W6,/W7     35,36,37,38,39,40,41,42  ;  OLMC'S   OUTPUTS

V2,V3,V4,V5,V6,V7,V8  25,26,27,28,29,30,31  ;  BLMC'S - Vertical counter
UNDERSIZE 32 ; BLMC

                    ;  S1   S2       state                    action

                    ;  0    0       idle                     clear W0-W7
                    ;  1    0       counting or send P       count W
                    ;  0    1       send W                   hold  W
                    ;  1    1       send V                   load V


EQUATIONS

W1 := /W1*S1*/S2 + V2*S1*S2 + S1*/S2*W1*W2*W3*W4*W5*W6*W7 ;
            /xW1V2 :=  [OR=S2]  [E=S1*/S2 + /S1*/S2 +S1*S2] [C=CLK]  ;
```